

تكنولوجيا المعلومات والاتصالات

(تصميم مواقع الويب وتطبيقات الموبايل)

الصف الثاني الثانوي

٢٠١٦/٢٠١٥



تكنولوجيا المعلومات والاتصالات

(تصميم مواقع الويب وتطبيقات الموبايل)

الصف الثاني الثانوي

إعداد

د/ طاهر عبد الحميد العدلي
خبير مناهج الكمبيوتروتكنولوجيا المعلومات
بمركز تطوير المناهج والمواد التعليمية

أ/ أحمد عبد الله منصور
خبير بالكمبيوتر التعليمي

أ/ محمد يوسف الصادق يوسف
خبير بالكمبيوتر التعليمي

أ/ أحمد الأنصاري السلاموني
خبير بالكمبيوتر التعليمي

مراجعة تربوية

د/ روجينا محمد حجازي
خبير المناهج بمركز تطوير المناهج والمواد التعليمية

مراجعة علمية

أ/ ماشاء الله محمد محمد
مدير عام الإدارة العامة للكمبيوتر التعليمي

أ.د/محمد فهمي طلبة
أستاذ الحاسبات والمعلومات بجامعة عين شمس

٢٠١٦/٢٠١٥

مقدمة الكتاب

عزيزي الطالب

تنبع أهمية البرمجة من كونها منهجًا للتفكير وأسلوبًا علميًا وعمليًا في حل العديد من المشكلات في القرن الحادي والعشرين الذي يطلق علية العصر الرقمي، وكونها أداة لتطوير مفردات هذا العصر.

ويعتمد المدخل الجديد لمنهج الكمبيوتر وتكنولوجيا المعلومات للصف الثاني الثانوي على تحقيق الأهداف العامة والأهداف الخاصة للمادة في ضوء وثيقة المنهج بما فيها من معايير ومؤشرات لهذا الصف، وذلك من خلال بناء المنهج على أساسيات المعرفة السابقة وتنمية مهارات التعلم الذاتي عن طريق تطبيق استراتيجية التعلم من خلال المشروعات، وهي إحدى الاستراتيجيات التربوية التي تنمي المهارات لدى المتعلمين عن طريق جمع المعلومات من مصادر متنوعة ومعالجتها واستخدامها في حل المشكلات من خلال عمل الفريق الواحد.

ويتكون محتوى الكتاب الإلكتروني التفاعلي من عدة موضوعات تتكامل فيما بينها لتحقيق تلك الأهداف المرجوة، حيث يتناول الكتاب أربع موضوعات أساسية هي: إنشاء موقع ويب تفاعلي متكامل من خلال لغة الترميز **HTML** أو من خلال إحدى التطبيقات، وإنشاء قاعدة بيانات وربطها بالموقع بالإضافة لإجراء معالجة على البيانات التي يتم عرضها على صفحات الموقع وما يتبعه من تأثير في البيانات المخزنة في قاعدة البيانات من خلال كود **Script**، وكل ذلك من خلال لغات برمجة مفتوحة المصدر وبرامج مجانية، بأسلوب يتيح للمتعلم فتح آفاقًا جديدة ورحبه تضعه على أول طريق توظيف ما يقوم بدراسته في حياته العلمية والعملية.

وقد تم توفير الكود بالكامل للمتعلم لإنشاء صفحات الموقع بحيث يكون التركيز على اكتساب المعارف والمهارات الأساسية باللغتين العربية والإنجليزية وأن يكون تركيز المتعلم على فهم وتطوير أي كود ويطور مهاراته في البحث والتقصي عنها بالمكتبات أو من خلال الإنترنت، كما يتضمن الكتاب صورًا من واجهات التطبيق الفعلية المنفذة للمشروع لتوضيح جميع خطوات العمل وتبسيط المفاهيم لدى المتعلمين وكل ذلك من خلال أنشطة وتدريبات عملية.

وأخيرًا فإن هذا الكتاب يحرص على تنمية المهارات الأساسية لدى الطلاب من خلال قضايا حياتية ليكون لديهم فرصًا يعبروا من خلالها عن إبداعاتهم ولتوضيح أهمية الكمبيوتر وتكنولوجيا المعلومات باعتبارها جزءً من حياتهم اليومية.

المؤلفون

محتوى الكتاب

أهداف كتاب الكمبيوتر وتكنولوجيا المعلومات والاتصالات للصف الثانى الثانوى العام:

- تعرف بعض المفاهيم (قواعد البيانات– لغات برمجة أنشاء صفحات ويب–تأمين مواقع الويب– برامج مفتوحة المصدر–برامج مجانية......) والعمليات والبنية الأساسية لنظم الكمبيوتر.
- تعرف بعض المفاهيم والعمليات الأساسية لخدمات وتطبيقات تكنولوجيا المعلومات والإتصالات.
- يُلم ببعض جوانب أمن المعلومات المتعلقة بإستخدام تكنولوجيا المعلومات والاتصالات
- يُلم ببعض جوانب الأمن الفكرى (الإنسانية والأخلاقية والاجتماعية) المتعلقة بإستخدام تكنولوجيا المعلومات والاتصالات.
- يستخدم أدوات الإنتاج التكنولوجية (HTML&PHP&SQL....) في دعم وتطوير تعلمه.
- ينتج أعمال ومشروعات ابدعية نسبيًا (مشروع قاموس مصطلحات الكمبيوتر المصور) باستخدام العمليات والبرامج والأدوات التكنولوجية.
- يوظف أدوات الأتصال التكنولوجية في التواصل والتفاعل والتعاون مع الآخرين لدعم التعلم.
- يوظف أدوات الاتصال التكنولوجية في تبادل المحتوى والرؤى مع الآخرين.
- يستخدم المصادر التكنولوجية في التعامل مع المعلومات الإلكترونية.
- يستخدم الأدوات التكنولوجية في معالجة البيانات وتقيميها وإعداد تقارير بالنتائج.

الوحدة الأولي

أساسيات تصميم مواقع الويب

فى نهاية الوحدة يتوقع أن يكون الطالب قادرًا على أن:

١- يوضح بعض المفاهيم والمصطلحات العلمية المتقدمة ذات الصلة بالكمبيوتر

٢- يوظف برامج الكمبيوتر في تنفيذ مهام التعلم.

٣- يوظف خدمات شبكة الإنترنت في عمليتى التعليم والتعلم.

٤- يمارس عمليات البحث عن المعلومات الإلكترونية بكافة صورها المتاحه على أجهزة الكمبيوتر وشبكات المعلومات الموثوقة في حل مشكلة.

مقدمة

استمرارًا لما بدأناه في مقرر الكمبيوتر للصف الأول الثانوي، والذي تناولنا فيه العديد من المعارف والمهارات المتنوعة من خلال مشروع "أطلس العربي الإلكتروني"، فإننا هذا العام –بإذن الله– سوف نحلق بكم في آفاقٍ جديدة وعالم مفتوح يحقق لكم المتعة والاستفادة والإبداع، وذلك من خلال تصميم وإنشاء موقع "قاموس مصطلحات الكمبيوتر المصور" باستخدام تطبيقات ولغات برمجة مجانية أو مفتوحة المصدر، والتي تَعُد الأكثر انتشارًا في كبرى الشركات والمؤسسات العالمية المتخصصة في تكنولوجيا المعلومات والاتصالات.

وخلال الفصل الدراسي الأول نسعى معًا لإنتاج موقع "قاموس مصطلحات الكمبيوتر المصور" والذي يَعُد بمثابة قالب تستطيع من خلاله إدخال كافة المصطلحات لأي مادة دراسية أخرى بحيث يسهل عليك البحث عن أي مصطلح وقراءة تعريفه بسرعة وسهولة وإمكانية تعديله أو حذفه.

الموضوع الأول

مدخل معرفى للمشروع

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يتعرف بعض المفاهيم والمصطلحات الأساسية المرتبطة بإنشاء موقع ويب يتعامل مع قواعد البيانات.

٢- يميز صفحة الويب الساكنة وصفحة الويب الديناميكية.

٣- يمارس عمليات إعداد جهاز الكمبيوتر ليصبح جهاز **Server**.

٤- يتعرف حزم برامج وتطبيقات الويب.

أثناء تنفيذك مشروع "قاموس مصطلحات الكمبيوتر المصور" ينبغي أن تتعرف بعض المفاهيم الاساسية الهامة التي سوف نستخدمها .

١-البرامج المجانية FreeWare

هي البرامج التي يسمح مالك البرنامج الأصلي "صاحب حق الملكية الفكرية" باستخدام برنامجه دون مقابل مادي أو إذن كتابي منه.

٢-البرامج مفتوحة المصدر Open Source Programs

هي البرامج والتطبيقات التي تُنشر ويسمح لمستخدميها بالاطلاع على الكود بحيث يمكنهم تعديل الكود وتطويره وإعادة نشره أو استخدامه بعد التعديل في ضوء احتياجات مطوريها.

٣-صفحة الويب الساكنة Static Web Page

هي صفحة المعلومات المتاحة على الإنترنت، والتي تعرض مختلف أنواع البيانات من نصوص وأرقام وصور وفيديو وجداول وارتباطات ... إلخ، وقد يصاحب بعض هذه البيانات تأثيرات معينة ويكون امتدادها **html**، والصفحة التالية مثال لصفحة **Static Web**:



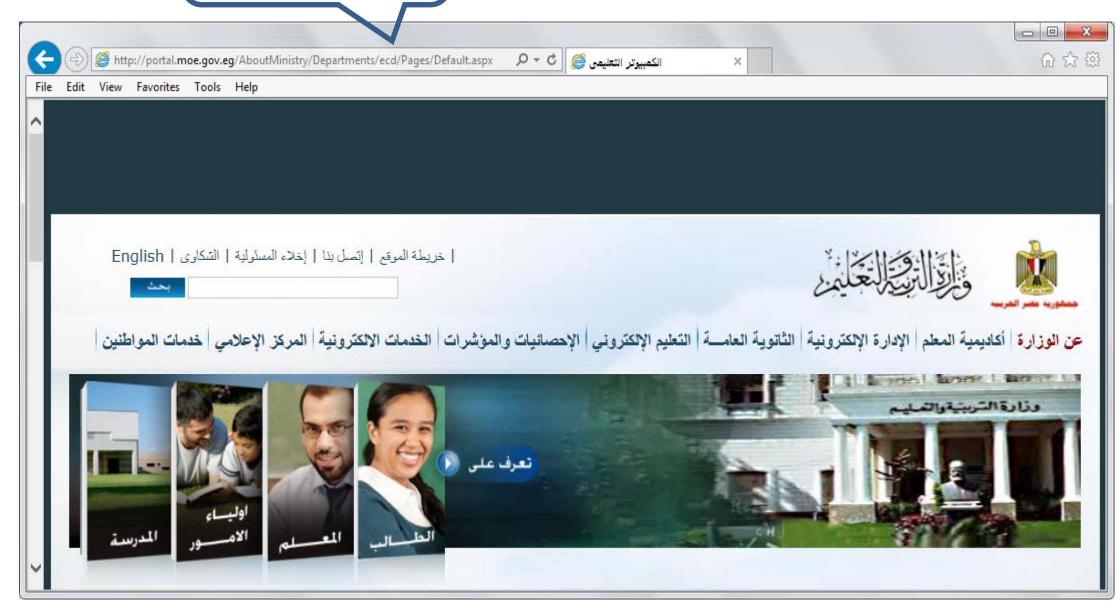
شكل (١) صفحة ويب ساكنة **Static Web Page**

٤-صفحة الويب الديناميكية Dynamic Web Page

هي صفحة المعلومات المتاحة على الإنترنت، والتي يمكن أن تعرض مختلف أنواع البيانات، حيث يتم من خلالها إجراء معالجة على بعض محتويات هذه الصفحة وإرجاع قيمة أو عرض رسالة أو ناتج، وتكتب الصفحة بلغة **PHP** أو **ASP.net** حيث تحتوى على كود ينفذ على جهاز الخادم **Server** ومنها على سبيل المثال:

- تنفيذ اجراء معين بناءًا على مدخلات العميل مثل اسم مستخدم وكلمة مرور.
- احتواء الصفحة على كود للتعامل مع قاعدة بيانات عند الخادم.

ويمكن في بعض الأحيان معرفة نوع الصفحة من الإمتداد في عنوان الموقع **URL** والشكل التالي مثال لصفحة **Dynamic Web**:



شكل (٢) صفحة ويب ديناميكية **Dynamic web page**

٥-الخادم Server

هذا المصطلح له معنيين:

الأول: قد يقصد به جهاز كمبيوتر خادم **Hardware Server** في شبكة كمبيوتر وهو جهاز يتميز بمواصفات فنية عالية عن باقي أجهزة الشبكة، يتحكم في باقي أجهزة الشبكة وتحدد من خلاله صلاحيات مستخدمي شبكة الكمبيوتر بواسطة نظام تشغيل الـ **Server**.

الأخير: يقصد به الدور الذي يقوم به جهاز معين في شبكة كمبيوتر **Software Server**، حيث نجد خادم الويب **Web Server** يقصد به جهاز الكمبيوتر المخزن عليه صفحات موقع الويب، أو خادم الطابعة **Print Server** ويقصد به جهاز الكمبيوتر المتصل به الطابعة بحيث يتحكم في إدارة عمليات الطباعة التي تصدر من أي جهاز آخر في الشبكة، أو خادم البريد الإلكتروني **Mail Server** وهو جهاز الكمبيوتر المُخزن عليه رسائل البريد الإلكتروني ويتحكم في إدارة كافة العمليات المرتبطة بالبريد الإلكتروني ويتيحها لباقي أجهزة الشبكة، أما خادم قاعدة البيانات **Database Server** فهو جهاز الكمبيوتر المُخزن عليه قاعدة البيانات ويتحكم ويدير عمليات تعامل باقي أجهزة الشبكة معها، .... وهكذا.

٦-سكريبت Script

كود يكتب بإحدى لغات البرمجة المخصصة لصفحات الويب لإجراء مهمة أو معالجة على بعض البيانات منها:

- **Client Side Languages** مثل **Java Script** و **VB Script** و ينفذ هذا الكود داخل المستعرض بجهاز العميل "**Run at Client**".
- **Server Side Languages** مثل PHP و **ASP.net** وينفذ هذا الكود داخل جهاز الخادم **Server** "**Run at Server**".

مثال : كود التأكد من عدم ترك حقل اسم المستخدم خالي يمكن عمله عن طريق **Java script** (**Run at Client**) أما كود التحقق من ان اسم المستخدم و كلمة السر صحيحة وهى الموجود في قاعدة البيانات بالخادم يتم بكود **PHP** أو **ASP** (**Run at Server**).

٧-لغة الترميز أو التكويد HTML "Hyper Text Markup Language"

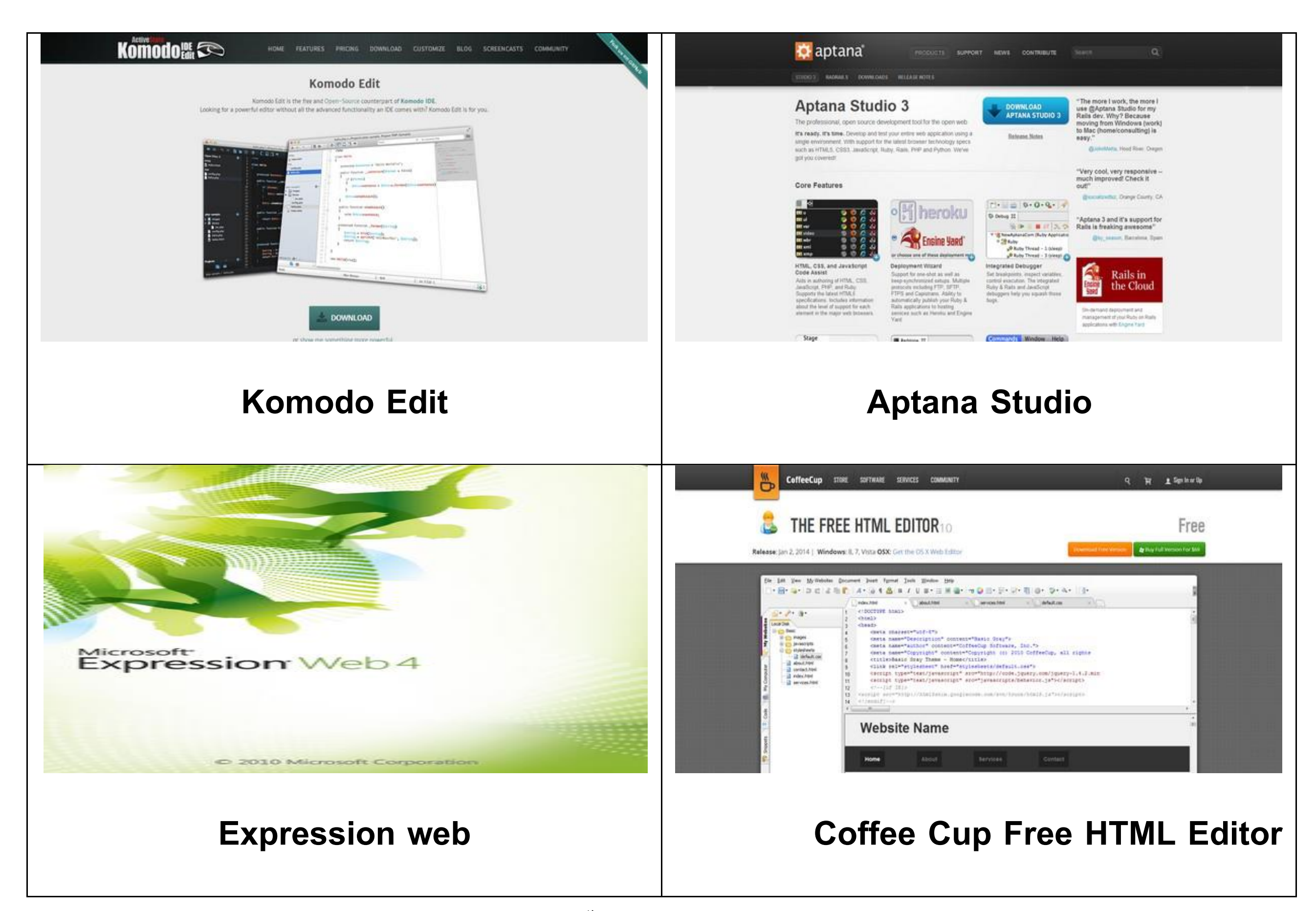
هي اللغة المستخدمة في إنشاء **Static Web Page**، ويمكن حفظ هذه الصفحات بامتداد **.htm, .html** وتعرض من خلال أحد مستعرضات الإنترنت.

٨-لغة PHP "Personal Home Paged"

إحدى اللغات المتخصصة في تطوير مواقع الويب الديناميكية، فهي لغة مفتوحة المصدر تتسم بالسهولة والسرعة وانها تعمل لدى الخادم **Server Side Language** ويتم تضمين الكود الخاص بها داخل كود **HTML**، كما أنها تحتوي على مكتبات ودوال تسهل التعامل مع ملفات **XML** وإرسال واستقبال الملفات بواسطة بروتوكول **FTP**، كما يمكنها الاتصال بسهولة بقواعد البيانات المختلفة بأمان وحماية.

وتُحفظ الصفحات التي تحتوي على كود **PHP** بامتداد **PHP**. ولتنفيذ كود **PHP** نحتاج لإجراء بعض التعديلات اللازمة لجعل جهاز الكمبيوتر الشخصي جهاز خادم **Server** باستخدام برنامج **Apache Server**.

وتحتاج أي لغة برمجة إلى برنامج مساعد يستخدم لكتابة الكود فيه ومن أشهر برامج محرر صفحات الويب بلغة **PHP**:



شكل (٣) بعض التطبيقات المستخدمة في إنشاء صفحات ويب

٩-برنامج Apache Server

يستخدم مع أجهزة الـ **Server** أو لتحويل جهازك الشخصي إلى جهاز خادم Server بحيث يستطيع ترجمة الكود المكتوب بلغة **PHP**، وهو ما يسهل على مطوري مواقع الويب من اختبار صفحات الموقع محليًا على جهاز الكمبيوتر **Local** قبل رفعه على الجهاز المُضيف **Host** ليتم نشره على الإنترنت.

١٠- نشر موقع ويب Publishing Web Site

عند إنشاء موقع ويب نحتاج إلى تجربة صفحاته وعرضه على مستعرض إنترنت وذلك قبل أن يكون متاحًا لزيارته على الإنترنت، وتوجد طريقتان لنشر موقع:

الأولى: محليًا **Localhost** حيث يتم عرض الموقع على جهاز الكمبيوتر الخاص بنا أو في شبكة محلية **LAN**.

الأخيرة: على الإنترنت **Publishing Web Site** حيث يُرفع الموقع على جهاز خادم يسمى جهاز **Hosting web site** أو الجهاز المُضيف حيث يسمح للمستخدمين بزيارة الموقع من خلال عنوان خاص بالموقع يطلق عليه **Web Address** أو **URL** "**Uniform Resource Locator**" مثل عنوان موقع وزارة التربية والتعليم "**www.emoe.org.eg**".

١١- MYSQL Server

هو أحد تطبيقات نظم إدارة قواعد البيانات العلائقية RDBMS "Relational Data Base Management System".

١٢- لغة SQL "Structured Query Language"

لغة برمجة تستخدم في كافة عمليات إدارة قواعد البيانات بدءً من إنشاء قاعدة البيانات والتعامل مع البيانات المخزنة داخل الجداول التي تتكون منها قواعد البيانات من خلال كتابة جمل بسيطة (أوامر) تسمح بإجراء عمليات:

- إضافة بيانات جديدة (**INSERT**).
- إستعراض بيانات مخزنة من قبل (**SELECT**).
- تعديل هذه البيانات (**UPDATE**).
- حذف هذه البيانات (**DELETE**).

١٣- حزم تطبيقات الويب "Web Server Pachages "

تشير حزم تطبيقات الويب إلى:

- تثبيت Install خادم الويب (**Apache Server**).
- تثبيت Install نظام لإدارة قواعد البيانات (**Mysql**).

ويتوفر على الإنترنت حزم تطبيقات ويب مجانية يمكن تثبيتها منها على سبيل المثال:

١- حزمة تطبيقات الويب LAMP "Linux, Apache, MySQL and PHP"

وتشمل برنامج خادم الويب **Server Apache**، وتطبيق قاعدة البيانات **MySQL**، ومترجم لغة البرمجة **PHP** وتعمل على نظام التشغيل **Linux**.

٢- حزمة تطبيقات الويب WAMP "Windows, Apache, MySQL and PHP"

وتشمل برنامج خادم الويب **Server Apache**، وتطبيق قاعدة البيانات **MySQL**، ومترجم لغة البرمجة **PHP** وتعمل على نظام التشغيل **Windows**.

٣- حزمة تطبيقات الويب MAMP "Mac, Apache, MySQL and PHP"

وتشمل برنامج خادم الويب **Server Apache**، وتطبيق قاعدة البيانات **MySQL**، ومترجم لغة البرمجة **PHP** وتعمل على نظام التشغيل **Mac**.

٤- حزمة تطبيقات الويب XAMPP "X-os, Apache, MySQL, Php , Perl"

وتشمل برنامج خادم الويب **Server Apache**، وتطبيق قاعدة البيانات **MySQL**، ومترجم لغة البرمجة **PHP** ولغة البرمجة **Perl** وتعمل على أي نظام تشغيل.

لاحظ أن الفرق الجوهري بين حزم تطبيقات الويب السابقة هو نظام التشغيل الذى تعمل معه.

تدريب ١ "مفاهيم أساسية" بكتاب الأنشطة والتدريبات

الموضوع الثاني

تخطيط مشروع موقع "قاموس مصطلحات الكمبيوتر المصور"

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يحدد المهام المطلوبة لإنشاء موقع ويب.

٢- يخطط الصفحة الرئيسية لموقع ويب

٣- يمارس تحميل برنامج **Xampp** لتشغيل تطبيقي **Apache** و **MYSQL**.

الهدف من المشروع

تصميم وإنشاء موقع "قاموس مصطلحات الكمبيوتر المصور" باستخدام تطبيقات ولغات برمجية مجانية أو مفتوحة المصدر.

أهمية المشروع

تكمن أهمية المشروع في تدريبك على كتابة كود PHP الخاص بالتعامل مع قواعد البيانات MySQL عبر صفحات موقع ويب

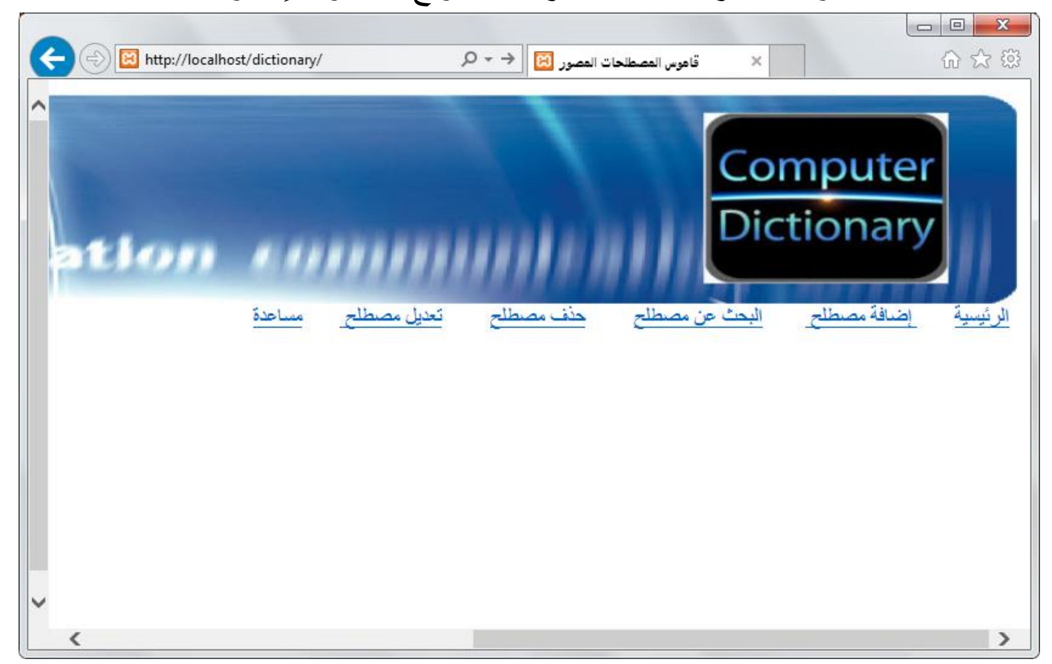
وبذلك يمكنك –عزيزي الطالب– اعتباره نموذجًا يمكنك تطبيقه فى انتاج مواقع جديدة متنوعة الإستخدامات بعد إجراءك التعديلات اللازمة.

تحديد المهام المطلوب تنفيذها فى مشروع إنشاء موقع "قاموس مصطلحات الكمبيوتر المصور"

- تهيئة **البيئة** المناسبة لإنشاء الموقع.
- **إنشاء قاعدة بيانات** لحفظ بيانات قاموس المصطلحات التي يتم إدخالها أو تعديلها عبر صفحات الموقع.
- **تصميم الصفحة الرئيسة.**
- **إضافة مصطلح** باللغتين العربية والإنجليزية – التعريف العلمي للمصطلح – صورة تدل على المصطلح، وحفظها في قاعدة بيانات.
- **البحث عن مصطلح** داخل قاعدة البيانات.
- **حذف مصطلح** من قاعدة البيانات.
- **تعديل بيانات مصطلح** في قاعدة البيانات.

- **توفير المساعدة** اللازمة حول كيفية التعامل مع الموقع.

الشاشة التالية تمثل نموذجًا مقترحًا للصفحة الرئيسية للموقع المطلوب إنشاؤه:



شكل (٤) مقترح الصفحة الرئيسية لموقع "قاموس مصطلحات الكمبيوتر المصور"

هل لديك مقترحات أخرى؟

ناقش زملائك ومعلمك فى مقترحاتك وامكانية تنفيذها.

اجراءات إنشاء موقع ويب

أولاً: تهيئة البيئة المناسبة لإنشاء الموقع وضبط ما يلزم من تقنيات لضمان سلامة تشغيل الموقع، ***ومن هذه الإعدادات والتقنيات التي يجب وضعها في الاعتبار:***

(١) تحديد البرامج التي سوف تستخدمها في بناء مشروعك وتثبيتها **Install** على الكمبيوتر.

(٢) تحديد طريقة حفظ ملفات صفحات معلومات الويب **Web Pages** الخاصة بالمشروع على جهاز كمبيوتر "خادم" **Server** ، وإمكانية عرضها على أحد برامج مستعرضات الإنترنت ، وتحفظ هذه الصفحات في صورة ملفات على جهاز كمبيوتر "خادم" **Server**، بإحدى الطرق التالية:

(أ) بمقابل مادي على الإنترنت من خلال حجز موقع على أحد أجهزة استضافة مواقع الويب " **Host Server**".

(ب) بدون مقابل على أحد مواقع الإنترنت المجانية التي تسمح لك باستضافة موقعك.

(ت) نشر الموقع محليًا على جهازك "**Local Host**"، وسوف نتبنى هذا الخيار –بإذن الله– في إنتاج مشروع "قاموس مصطلحات الكمبيوتر المصور".

☺ **(ملحوظة هامة:)** ☺ يفضل وضع جميع الملفات الخاصة بالموقع في مجلد واحد يحتوي على مجلدات فرعية تضم الملفات ذات النوع الواحد معًا، وأن تكون أسماء الملفات مختصرة ومعبرة عن محتواها وباللغة الانجليزية.

(٣) إعداد جهاز الكمبيوتر الخاص بك ليقوم بدور جهاز الخادم المضيف للموقع "**Local Host**":

عند إنشاء موقع ويب يحتاج مطوري البرامج إلى تهيئة البيئة المناسبة لإنشاء موقع الكتروني "قاموس مصطلحات الكمبيوتر المصور" مما يستلزم جعل جهازك يعمل كجهاز خادم Server، وذلك باستخدام برنامج Apache Server الذي يقوم بترجمة الكود المكتوب بلغة **PHP** "**Personal Home Pages**" واختبار الموقع على جهازك قبل نشره على الإنترنت أسوة بما تقوم به الشركات الكبرى.

ولتجربة الموقع والتأكد من صحته محليًا قبل نشره على الإنترنت أسوة بما تقوم به الشركات الكبرى، تحتاج إلى تهيئة جهازك ليقوم بدور جهاز الخادم Server ليصلح لاستضافة الموقع.
ومن أمثلة البرامج المستخدمة لهذا الغرض برنامج "Xampp "
يعتبر برنامج Xampp من أفضل برامج إعداد جهاز الكمبيوتر "Local Host"، فهو يتسم بالسهولة والقدرة على التعامل مع قواعد البيانات MYSQL

ثانياً: تنفيذ مشروع موقع "قاموس مصطلحات الكمبيوتر المصور" :

وتمر بثلاث مراحل ، هى:

(١) مرحلة انشاء قاعدة بيانات: تخزن فيها البيانات التي يتم إدخالها أو عرضها على صفحات موقع الويب، وذلك من خلال إحدى برامج إدارة قواعد البيانات مثل: MYSQL – Access .... إلخ.

(٢) مرحلة انشاء صفحة/موقع ويب ساكنة Static Web Pages وذلك من خلال أحد الخيارين التاليين:

- لغة الترميز HTML "Hyper Text Markup Language".
- استخدام برنامج جاهز أو تطبيق Application مثل برنامج: Expression Web

لاحظ :

الموقع الذي يتم إنشاؤه بإستخدام تطبيق أو لغة HTML يحتوي على صفحات ويب ساكنة "STATIC WEB PAGES" تضم صورًا مختلفة من البيانات والمعلومات.

(٣) مرحلة تحويل صفحة الويب الساكنة إلى صفحة ويب ديناميكية "Dynamic Web Pages":

يتم فيها معالجة محتويات صفحة الويب الساكنة والتعديل في قاعدة البيانات المرتبطة بها بواسطة المستخدم المسموح له بالمعالجة والتعديل، وذلك من خلال كتابة كود بإحدى لغات البرمجة مثل: PHP داخل كود HTML.

(٤)عرض صفحات الموقع بإستخدام أحد مستعرضات الإنترنت: مثل: Internet Explorer – Google Chrome – Fire Fox ...إلخ.

نشاط ١: إعداد جهاز الكمبيوتر ليصبح جهاز Server بتثبيت install برنامج "Xampp" .
تحت إشراف معلمك اتبع الخطوات الموضحة فى كتاب الأنشطة والتدريبات لتثبيت برنامج Xampp

عزيزى الطالب/عزيزتى الطالبة لإنشاء مشروعنا بموقع "قاموس مصطلحات كمبيوتر مصور" علينا ما يلى:

١- إنشاء صفحات ساكنة "Static Web". وهو ما يتم من خلال تطبيق يمكن من خلاله عرض كود "HTML" مثال تطبيق"Expression Web"

٢- معالجة محتويات الصفحات الساكنة "Static Web" لتصبح صفحات الموقع Dynamic ، وذلك بإضافة كود بلغة "PHP" داخل كود "HTML".

٣- تشغيل برنامج Xampp الذي يقوم بالآتي:

تشغيل برنامج Appache والذى يجعل الجهاز يقوم بدور الخادم Server

تشغيل برنامج MYSQL والذى يمكن من التعامل مع قواعد البيانات

<u>وهو ما ستقوم بتنفيذه تحت اشراف معلمك ،من خلال اتباعك لإجراءات الأنشطة الموضحة بكتاب الأنشطة والتدريبات)</u>

الوحدة الثانية

متطلبات ومراحل إنتاج المشروع

فى نهاية الوحدة يتوقع أن يكون الطالب قادرًا على أن:

١- يقترح مشروعات بسيطة لتوظيف تطبيقات وخدمات تكنولوجيا المعلومات والاتصالات في المجالات الحياتية والدراسية

٢- يوظف تطبيقات تكنولوجيا المعلومات والأتصالات في تبادل مهام المشروعات

٣- يوظف البرامج والتطبيقات المكتبية(جداول حسابية-قواعد البيانات) في إنتاج مهام محددة ضمن مشروعات تعليمية.

٤- يوظف عدة برامج لحل المشكلات التعليمية والحياتية.

٥- يصمم قاعدة بيانات مناسبة لمشروعه

٦- يمارس مهارات البرمجة بلغة **PHP**.

٧- يستخدم بعض تطبيقات الانترنت فى مشروعه.

٨- يصمم صفحات ويب تفاعلية خاصة بمشروعه.

٩- يربط صفحات الويب التفاعلية بقواعد البيانات.

١٠- ينتج مشروع باستخدام تطبيقات الانترنت المتقدمة.

١١- يوظف قواعد البيانات المتقدمة (**MySQL**) فى تصميم مواقع لحل مشكلة حياتية.

١٢- يستخدم لغة البرمجة **PHP** فى انتاج برمجيات وموقع ويب للتعبير عن افكاره.

١٣- يقيم المتعلم مواقع ومصادر المعلومات من حيث الدقة والمصداقية في ضوء معايير محددة.

عزيزى الطالب والطالبة يمر مشروع إنتاج "قاموس مصطلحات الكمبيوتر المصور " بعدة مراحل أساسية هي:

١- مرحلة تصميم صفحات الموقع

٢- مرحلة إنشاء جداول قاعدة البيانات

٣- مرحلة إنشاء صفحات الموقع

وكل مرحلة تتضمن عدة مهارات ومن خلال دراستك لها وتدريبك عليها يمكنك تنفيذ المشروع

الموضوع الأول

تصميم صفحات الموقع

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم مكونات الصفحة الرئيسية لصفحات موقع قاموس المصطلحات.

٢- يحدد الروابط اللازمة للإبحار بين صفحات الموقع.

٣- يناقش مع زملاءه عدة مقترحات لصفحات مشروع.

مرحلة التصميم هى المرحلة الأولى من مراحل تنفيذ المشروع ، وفيها يتم تحديد صفحات الويب، واسمها، ومحتواها، والغرض من كل صفحة، ووضع تصور أو تصميم لمكونات الصفحة الرئيسة للموقع والارتباطات التي نستخدمها في التنقل بين صفحات الموقع والعودة مرة أخرى للصفحة الرئيسة، والشكل التالي يوضح صفحات موقع "قاموس مصطلحات الكمبيوتر المصور":



صفحة إدخال مصطلح

Add_Term.php



صفحة المساعدة

Help.php



الصفحة الرئيسة

Index.php



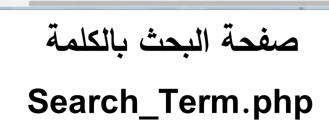
صفحة حذف مصطلح

Del_Term.php



صفحة تعديل مصطلح

Edit_Term.php



صفحة البحث بالكلمة

Search_Term.php

شكل (٥) مخطط لصفحات موقع قاموس "مصطلحات الكمبيوتر المصور"

الجدول التالي يعرض صفحات موقع "**قاموس مصطلحات الكمبيوتر المصور**" والغرض منها، وأهم الملاحظات عليها :

م	اسم الصفحة	الغرض منها	ملاحظات
١	صفحة الاتصال بقاعدة البيانات **Connection.php**	تحقيق الاتصال بقاعدة البيانات، حيث يتم استدعائها قبل التعامل مع البيانات.	لا تظهر هذه الصفحة أمام المستخدم وإنما تم فصلها لكي نكتفي بسطر كود واحد يحقق الاتصال بقاعدة البيانات في بداية كل صفحة بدلا من تكرار كود الاتصال بقاعدة البيانات بالكامل.
٢	صفحة رأس صفحات الموقع **Header.php**	إظهار الصورة والارتباطات التي تتنقلا لجميع صفحات الموقع حيث يتم استدعائها في بداية جميع صفحات الموقع.	هذه الصفحة تحتوي على الأجزاء الثابتة التي نحتاج لعرضها في جميع صفحات الموقع، لذا تم فصلها في صفحة مستقلة، وتستدعى بسطر كود واحد في بداية كل صفحة.
٣	صفحة إضافة مصطلح **Add_Term.php**	إدخال مصطلح وجميع بياناته في جدول المصطلحات بقاعدة البيانات.	
٤	صفحة البحث عن مصطلح **Search_Term.php**	البحث عن مصطلح في جدول المصطلحات بقاعدة البيانات.	
٥	صفحة تعديل مصطلح **Edit_Term.php**	تعديل بيانات مصطلح في جدول المصطلحات بقاعدة البيانات.	عند إجراء تعديل بيانات مصطلح نختار المصطلح المراد تعديله، ثم إجراء ما يلزم من تعديل وحفظها.
٦	صفحة حذف مصطلح **Del_Term.php**	حذف بيانات مصطلح في جدول المصطلحات بقاعدة البيانات.	يمكن حذف المصطلح الذي تم اختياره.
٧	صفحة المساعدة **Help.php**	عرض معلومات عن الموقع وتبعيته.	عادة ما نجد في معظم المواقع صفحة **About Us** يعرض فيها معلومات عن الجهة التي يتبعها الموقع وطرائق التواصل مثل أرقام الهواتف والعنوان والبريد الإلكتروني والعنوان على

م	اسم الصفحة	الغرض منها	ملاحظات
			مواقع التواصل الاجتماعي.
٨	الصفحة الرئيسة **Index.php**	تفعيل الارتباطات التي يمكن من خلالها الانتقال بين صفحات الموقع.	تمثل صفحة البداية التي تظهر للمستخدم عند تحميل الموقع، والتي من خلالها يمكن الانتقال إلى باقي صفحات الموقع.

مرحلة التصميم الجيد للموقع قبل البدء في إنشائه يوفر الكثير من الوقت والجهد ويساعد على إنتاج موقع بشكل احترافي متميز،

وفى هذه المرحلة وبدراسة صفحات الموقع نلاحظ أن بها جزء ثابت لا يتغير وهو:

- الصورة التي تظهر في بداية كل صفحة وتسمى **Banner**
- جميع الارتباطات التشعبية **Hyperlinks** التي نستخدمها في الإبحار بين صفحات الموقع وهي: (الرئيسة – إدخال مصطلح – بحث عن مصطلح – تعديل – حذف – مساعدة)،

وهذا يتطلب تكرار الكود الخاص بالصورة والارتباطات فى جميع صفحات الموقع وهو ما يمثل عبء كبير وتكرار ينبغي أن نتجنبه، وذلك بفصلها في صفحة واحدة يتم استدعائها في باقي الصفحات

ويمكن حل هذه المشكلة ببساطة بإضافة صفحة باسم "**Header.php**" تحتوي على الصورة **Banner**، **والارتباطات التتشعبية Hyperlinks** "أسماء الصفحات أو الخدمات التي يتيحها الموقع."

والجدير بالذكر أننا نستطيع تنفيذ إنشاء صفحات الويب الخاصة بالمشروع من خلال العديد من التطبيقات أو البرامج الجاهزة بسرعة وسهولة، وسوف تدرس تلك المهارات فى المراحل التالية .

بالإضافة إلى تحديد صفحات الموقع، علينا تحديد الهيكل البنائى لقاعدة البيانات، وهو ما سيتم توضيحه فى المرحلة الثانية

تدريب: تحليل مخطط لموقع

الغرض من كل صفحة (اذكر اكبر عدد من الأفكار...........................)

الموضوع الثاني

إنشاء قاعدة البيانات

فى نهاية الموضوع ينبغى أن تكون قادرًا على أن:

١- يتعرف بعض المفاهيم الأساسية المتعلقة بقواعد البيانات.

٢- ينشيء قاعدة بيانات لمشروع قاموس مصطلحات الكمبيوتر.

٣- يمارس عمليات الربط بين قواعد البيانات.

٤- يصنف العلاقات فى قواعد البيانات.

٥- يستخدم برنامج **MYSQL** فى إنشاء قاعدة بيانات.

٦- يتعاون مع زملاءة فى كتابة انشاء قاعدة البيانات بلغة البرمجة.

المرحلة الثانية فى المشروع هى مرحلة إنشاء قاعدة بيانات مشروع "قاموس مصطلحات الكمبيوتر المصور" وقبل البدء لابد من التعرف على بعض المفاهيم الأساسية الخاصة بقواعد البيانات.

١-قاعدة البيانات Database:

قاعدة البيانات عبارة عن تخزين أو حفظ مجموعة من البيانات المنظمة والمرتبطة بموضوع معين بغرض استرجاعها لاتخاذ القرارات. وتعتبر قاعدة البيانات العلائقية Relaional database احد أنواع قواعد البيانات التى تعتمد على تقسيم البيانات فى جداول Tables مع تحديد العلاقات بين هذه الجداول.

٢-الجداول Tables

الجدول يمثل البنية الأساسية أو المكون الأساسي لقاعدة البيانات، ويتكون من سجلات **Records**، وحقول **Fields** مثل: جدول بيانات: الطالب أو الموظف أو المنتج...إلخ.

اسم الطالب	نوع الطالب	المحافظة	تاريخ الميلاد	التليفون
محمد	ذكر	القاهرة	٢٠٠٠/١/١	
سما	أنثى	الجيزة	١٩٩٩/٢/٥	
فاطمة	أنثى	القاهرة	٢٠٠٢/٤/٣	
يوسف	ذكر	الجيزة	٢٠٠١/٥/١١	
هبه	أنثى	الشرقية	١٩٩٨/٧/٦	

شكل (٦) مثال لجدول طالب

من الجدول بالشكل السابق نستنتج المفاهيم التالية:

أ- السجلات Records:

عبارة عن صف بالكامل من جدول البيانات يحتوي على جميع البيانات الخاصة بشخص واحد فقط أو حالة واحدة، ويتكون من عدة حقول مثل سجل بيانات (طالب أو موظف أو منتج معين) مثل سجل بيانات: الطالب محمد أو الموظف سامح أو المنتج أو الصنف كمبيوتر ...إلخ.

ب- الحقول Fields:

ويمثل الحقل البنية الأساسية التي يتكون منها جدول البيانات ، أي عمود في جدول يمثل حقل ويحتوي على بيان واحد فقط لكل سجل من سجلات الجدول، .

لكل حقل العديد من الخصائص منها:

- اسم الحقل Field Name مثل: اسم الطالب، اسم المحافظة، المرتب، الكمية ...إلخ.
- نوع الحقل Field Type : قد تكون حقل نصstring، مثل: الاسم أو العنوان ، حقل رقمى numeric مثل: درجة أو راتب موظف، وحقل تاريخ date ، مثل: حقل تاريخ الميلاد وهكذا.
- حجم الحقل Field Size وهو عدد الخانات أو الأحرففى حالة الحقل النصى ، أو تحديد نوع الاقام المدخلة (صحيح Integer – يقبل الكسر العشرى decimal – .... الخ) فى حالة الحقل الرقمى

والجدير بالذكر أنه يمكن **إنشاء قاعدة بيانات تحتوي على جدول واحد فقط** يحتوى على جميع الحقول اللازمة وفي هذه الحالة يسمى هذا الجدول "**Flat Table**".

ولكن عادة ما نقوم بفصل الحقول التي يجمعها عامل مشترك في جدول واحد، **وبالتالي تحتوي قاعدة البيانات على أكثر من جدول.**

وهو الأمر الذي يتطلب إنشاء **علاقات تربط جداول قاعدة البيانات ببعضها** بحيث نتجنب تكرار إدخال البيانات أو الحقول، ويطلق عليها في هذه الحالة **قاعدة بيانات علائقية Relational Database**..

وحتى تتم عملية الربط بين جداول قاعدة البيانات بنجاح، فلابد من تحقيق الآتي:

(١) تعيين حقل مفتاح أساسي لكل جدول يسمى "**Primary Key**"، وذلك بأحد طريقتين هما:

- **باختيار أحد حقول الجدول** بشرط ألا يتكرر أي بيان في هذا الحقل، حتى مع حجم ضخم من البيانات
- **أو بإضافة حقل جديد** يتم تعينه "حقل مفتاح أساسي".

ناقش مع زملائك ومعلمك تدريب (١)، بكتاب الأنشطة والتدريبات ص ...

ومنه يتضح أن الحقول التى يمكن تكرار بياناتها لا تصلح أن تكون حقل مفتاح أساسي، ، لذا يجب إضافة حقل جديد يقوم بدور مفتاح أساسي للجدول ، وليكن **رقم الطالب أو كود الطالب** بحيث يخصص لكل طالب رقم خاص به يميز بياناته عن باقي سجلات الجدول.

(٢) تحديد أنواع العلاقات Relationships في قواعد البيانات، ومفهوم كل نوع، وكيف يمكن الربط بين جدولين، وهو ما سيتم تناوله بشيء من التفصيل.

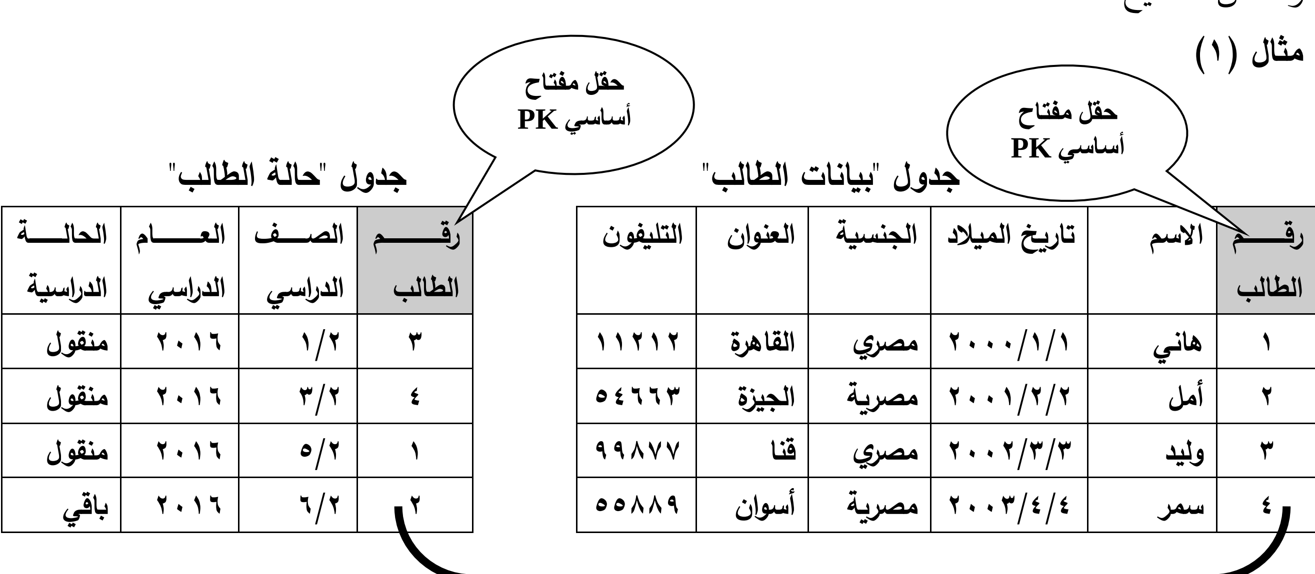
٣- أنواع العلاقات في قواعد البيانات

يوجد ثلاثة أنواع من العلاقات للربط بين جداول قاعدة البيانات:

النوع الأول: علاقة رأسٍ برأس One to One

علاقة تربط بين جدولين بحيث يمكن أن يرتبط سجل في الجدول الأول بسجل واحد وواحد فقط من الجدول الثاني والعكس صحيح.

مثال (١)



جدول "بيانات الطالب"

رقم الطالب	الاسم	تاريخ الميلاد	الجنسية	العنوان	التليفون
١	هاني	٢٠٠٠/١/١	مصري	القاهرة	١١٢١٢
٢	أمل	٢٠٠١/٢/٢	مصرية	الجيزة	٥٤٦٦٣
٣	وليد	٢٠٠٢/٣/٣	مصري	قنا	٩٩٨٧٧
٤	سمر	٢٠٠٣/٤/٤	مصرية	أسوان	٥٥٨٨٩

جدول "حالة الطالب"

رقم الطالب	الصف الدراسي	العام الدراسي	الحالة الدراسية
٣	٢/١	٢٠١٦	منقول
٤	٢/٣	٢٠١٦	منقول
١	٢/٥	٢٠١٦	منقول
٢	٢/٦	٢٠١٦	باقي

شكل (٧) يوضح العلاقة رأس برأس بين جدولين

من الشكل (٧) نلاحظ أن الربط بين جدولين بعلاقة رأس برأس تتم بربط حقل المفتاح الأساسي في الجدول الأول "رقم الطالب" مع حقل المفتاح الأساسي في الجدول الثاني "رقم الطالب"، مع مراعاة أن حقلي المفتاح الأساسي في الجدولين لابد أن يكونا:

(١) من نفس نوع البيانات **Field Data Type**.

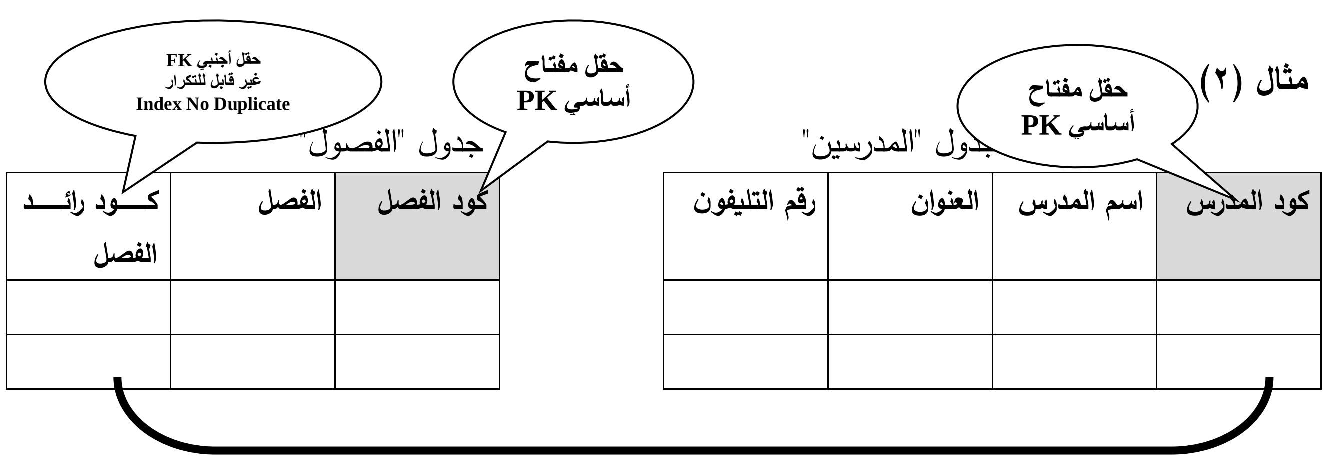
(٢) نفس الحجم.

(٣) نفس المحتوى.

ولا يشترط أن يكونا نفس الاسم.

يقوم تطبيق قاعدة البيانات بربط سجل رقم الطالب (١) في الجدول الأول بسجل رقم الطالب (١) في الجدول الثاني وهكذا لباقي السجلات حتي وإن كان إدخال السجلات في الجدولين بترتيب مختلف، المهم أن يكون محتوى حقلي الربط متماثلين.

مثال (٢)



شكل (٨) يوضح حالة أخرى لعلاقة رأس برأس بين جدولين

من الشكل (٨) نلاحظ أنه يُحدد لكل فصل رائد واحد فقط من المدرسين، حيث يرتبط حقل كود المدرس **(PK)** في جدول "المدرسين" بالحقل الأجنبي كود رائد الفصل **(FK)** في جدول "الفصول" على أن يتم ضبط خصائص حقل كود الرائد غير قابل للتكرار "**Index No Duplicate**".

ملحوظة: يطلق على حقل كود رائد الفصل (FK) حقل أجنبي" FOREIGN KEY لأنه ليس من طبيعة الجدول بل حقل دخيل عليه.

النوع الثاني: علاقة رأس بأطراف One to Many

علاقة تربط بين جدولين بحيث يمكن أن يرتبط سجل في الجدول الأول بعديد من السجلات في الجدول الثاني والعكس غير صحيح.

مثال:

حقل مفتاح
أساسي PK

جدول الصف

رقم الصف	اسم الصف	مكان الصف
١٠١	الأول	الدور الأول
٢٠١	الثاني	الدور الثاني
٣٠١	الثالث	الدور الثالث

حقل مفتاح
أساسي PK

حقل أجنبي
FK

جدول الطلاب

رقم الطالب	العام الدراسي	الشعبة	رقم الصف
٣	٢٠١٦	علمي	١٠١
٤	٢٠١٦	علمي	١٠٣
١	٢٠١٦	أدبي	١٠١
٢	٢٠١٦	علمي	١٠٢
٥	٢٠١٦	أدبي	١٠٢
٧	٢٠١٦	علمي	١٠٣
٩	٢٠١٦	أدبي	١٠٣
١٠	٢٠١٦	أدبي	١٠١
١٢	٢٠١٦	أدبي	١٠١

شكل (٩) يوضح العلاقة رأس بأطراف بين جدول "الصف" وجدول "الطلاب"

من الشكل (٩) نلاحظ الآتي:

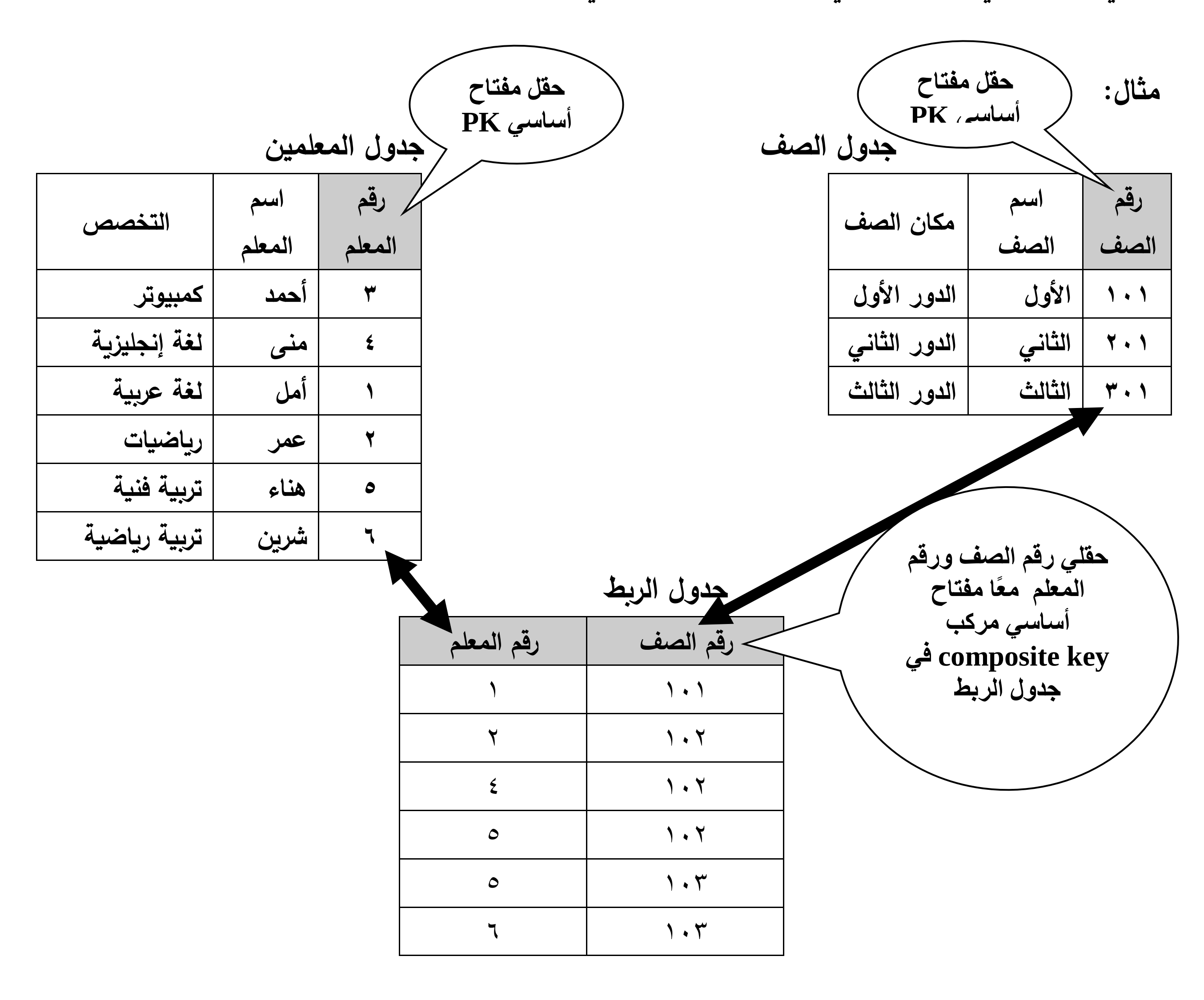
- العلاقة بين جدولي "الصف" و "الطلاب" علاقة رأس بأطراف من ناحية جدول الصف، حيث نجد أن كل سجل في جدول "الصف" يرتبط به عديد من السجلات في جدول "الطلاب" أي أن الصف الواحد يوجد به عديد من الطلاب في حين أن الطالب لا ينتمي إلا لصف واحد.
- للربط بين الجدولين في علاقة رأسٍ بأطراف ينبغي إضافة حقل المفتاح الأساسي في جدول **One** وهو جدول "الصف" إلى جدول **Many** وهو جدول "الطلاب" ويسمى في هذه الحالة "حقل أجنبي" **Foreign Key** لأنه ليس من طبيعة الجدول بل حقل دخيل عليه.
- في الحقل الأجنبي يمكن تكرار البيانات (فى العلاقة رأس بأطراف).
- أن كل سجل في جدول "الصف" يرتبط بعديد من السجلات في جدول "الطلاب".

يقوم تطبيق قاعدة البيانات بربط كل سجل في جدول الرأس "الصف" بسجلات الطلاب التي أمامها نفس رقم الصف في الحقل الأجنبي في جدول أطراف "الطلاب".

النوع الثالث: علاقة أطراف بأطراف Many to Many

علاقة تربط بين جدولين بحيث يمكن أن يرتبط سجل في الجدول الأول بعديد من السجلات في الجدول الثاني، وسجل في الجدول الثاني بعديد من السجلات في الجدول الأول.

مثال:



جدول الصف

رقم الصف	اسم الصف	مكان الصف
١٠١	الأول	الدور الأول
٢٠١	الثاني	الدور الثاني
٣٠١	الثالث	الدور الثالث

جدول المعلمين

رقم المعلم	اسم المعلم	التخصص
٣	أحمد	كمبيوتر
٤	منى	لغة إنجليزية
١	أمل	لغة عربية
٢	عمر	رياضيات
٥	هناء	تربية فنية
٦	شرين	تربية رياضية

جدول الربط

رقم الصف	رقم المعلم
١٠١	١
١٠٢	٢
١٠٢	٤
١٠٢	٥
١٠٣	٥
١٠٣	٦

١٠١	٣
١٠١	٥
١٠٣	٢

شكل (١٠) يوضح العلاقة أطراف بأطراف بين جدول "الصف" وجدول المعلمين

نلاحظ من الشكل (١٠) الآتي:

- العلاقة بين جدولي "الصف" و"المعلمين" علاقة أطراف بأطراف، حيث نجد أن كل سجل في جدول "الصف" يمكن أن يرتبط به عديد من السجلات في جدول "المعلمين"، وأي سجل في جدول "معلمين" يمكن أن يرتبط بعديد من السجلات في جدول "الصف"، أي أن الصف الواحد يمكن أن يُدَرس فيه أكثر من مُعلم، وأيضًا يمكن للمعلم الواحد أن يقوم بالتدريس في أكثر من صف.
- قبل الربط بين الجدولين في علاقة أطراف بأطراف ينبغي إنشاء جدول جديد يحتوي على حقل المفتاح الأساسي في جدول الصف وهو "رقم الصف"، وحقل المفتاح الأساسي في جدول المعلمين وهو "رقم المعلم"، وأن يتم تعيين الحقلين معًا مفتاح أساسي في جدول الربط.
- العلاقة أطراف بأطراف هي علاقة نظرية في مفاهيم قواعد البيانات، ولا يمكن تمثيلها عمليًا في برامج قواعد البيانات.
- لذا تم تكسير العلاقة إلى علاقتين:

الأولى: علاقة رأس بأطراف بين جدول "الصف" وجدول الربط بحيث يكون الرأس هو جدول "الصف"، والأطراف من ناحية جدول الربط.

والثانية: علاقة رأس بأطراف بين جدول "المعلمين" وجدول الربط بحيث يكون الرأس هو جدول "المعلمين"، والأطراف من ناحية جدول الربط.

لاحظ:

- **لا يمكن تكرار البيانات في حقل المفتاح الأساسي في جدول "الصف"**
- **لا يمكن تكرار البيانات في حقل المفتاح الأساسي في جدول "المعلمين**
- **لا يمكن تكرار رقم معلم ورقم الصف معًا في جدول الربط.**
- **كل صف في جدول "الصف" يمكن أن يدخله عديد من المعلمين، وأيضًا أن كل معلم يمكن أن يدخل أكثر من صف دراسي.**

٤- يمكن إنشاء قاعدة بيانات من خلال كود برمجة كما يلي:

استخدام لغة البرمجة SQL في PHP وكتابة الكود الخاص بإنشاء:

- قاعدة البيانات بالأمر **Create Database DB-TEST**، ويشير الجزء المُظلل إلى اسم قاعدة البيانات.
- جدول بيانات بالأمر **Create Table Table-Test**، ويشير الجزء المُظلل إلى اسم جدول قاعدة البيانات.

- حقول جدول البيانات بالأمر **Col1–Name varchar(15)**، ويشير الجزء المُظلل إلى اسم الحقل ونوعه وحجمه، ويمكن تكرار هذا الكود لأي عدد من الحقول مع تغيير خصائص الحقل.

٥- استخدام أحد تطبيقات نظم إدارة قواعد البيانات مثل: Oracle أو MS Access أو MySQL،

وسوف نبدأ أولى خطوات إنشاء قاعدة بيانات المشروع باستخدام **MYSQL** لأسباب عديدة من أهمها:

- أنها مجانية **Free Ware**.
- مفتوحة المصدر **Open Source**.
- تسمح بالتعامل مع كم هائل من البيانات.
- تستخدمه كبرى الشركات العالمية خاصة في مجال تكنولوجيا المعلومات مثل: **Google, Yahoo and Friendstar** ولمزيد من المعلومات حول بعض الشركات التي تستخدم **MYSQL** يمكن زيارة الموقع التالي:

http://www.mysql.com/customers/

عادة ما تضم التطبيقات المستخدمة في نظم إدارة قواعد البيانات كائنات أساسية من أهمها:

- **جداول Tables:** من خلالها يمكن إنشاء جداول قاعدة البيانات بما تحويه من سجلات وحقول وإمكانية إدخال بيانات فيها وعرضها وتعديلها.
- **استعلامات Queries:** الاستعلام عباره عن سؤال يوجهه المستخدم لقاعدة البيانات فيعرض البيانات التي تجيب على هذا السؤال، ومن خلال الاستعلام يمكن عرض بعض حقول من جدول بيانات أو يعرض بعض سجلات من جدول أو أكثر بناءً على شرط معين ...إلخ.

لإنشاء قاعدة بيانات مشروع "قاموس مصطلحات الكمبيوتر المصور" باستخدام تطبيق MYSQL. اتبع اجراءات نشاط "إنشاء قاعدة بيانات باستخدام برنامج MySQL " بكتاب الأنشطة والتدريبات ص

نشاط ١ : إنشاء قاعدة بيانات باستخدام برنامج MySQL

يساعدك تنفيذ هذا النشاط على إنشاء قاعدة بيانات dbdictionary باستخدام تطبيق/برنامج MySQL.
ويعتبر هذا النشاط مرشدًا يمكنك تحت اشراف معلمك من إنشاء أي جداول أخرى في قاعدة البيانات حسب احتياجاتك.

بالمواصفات التالية:

مواصفات قاعدة بيانات المشروع :

اسم قاعدة اليانات: dbdictionary ، تحتوي جدول بيانات باسم Terms، يضم عدد (٥) خمسة حقول كما هو موضح بالجدول التالي:

Table: Terms				
#	Name	Type	Length values	Index
1	Id	Integer	11	Primary
2	Term	VarChar	50	
3	Trans	VarChar	500	
4	Defe	VarChar	5000	
5	Picture	VarChar	300	

ملاحظة : تم انشاء قاعدة بيانات من جدول واحد للتبسيط وخصوصا عند البرمجة بلغة PHP

الموضوع الثالث

إنشاء صفحات الموقع

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يمارس خطوات إنشاء موقع بإستخدام تطبيق **Expression Web**.

٢- يتعرف لغة الترميز **HTML**.

٣- ينشيء صفحة ويب بكود **HTML**.

٤- يشرح مكونات الشاشة الرئيسية لبرنامج **Expression Web**.

٥- يربط بين صفحات الويب بكود **HTML**.

٦- يتعرف لغة البرمجة **PHP**.

٧- يوظف إحدى تطبيقات إنشاء مواقع ويب فى الربط بين صفحات موقع ويب.

قبل البدأ فى انشاء صفحات الموقع ، علينا تنظيم الملفات التى سيتم عرضها على صفحات الموقع ، والتي ينبغي أن تكون في مجلد واحد، حيث يحتوي هذا المجلد على مجلدات فرعية لجميع المعلومات والبيانات والملفات التي يتم عرضها على صفحات الموقع، مثل مجلد لملفات الصور ... إلخ.

والجدير بالذكر أنه عند تثبيت برنامج **Xampp**، فقد تم تثبيته بشكل افتراضي داخل مجلد باسم **Xampp** على مشغل الأقراص :C، وبه مجموعة من المجلدات منها مجلد باسم **Htdocs**، وينبغي إنشاء مجلد للمشروع باسم "**Dictionary**" داخل المجلد **Htdocs** ليصبح المسار كما يلي:

C:\xampp\htdocs\dictionary

وهو المسار الذي سوف تستخدمه في حفظ جميع صفحات الموقع ، وحفظ المجلدات الفرعية لجميع الملفات المتعلقة بالموقع.

اجراءات اعداد ملفات ومجلدات المشروع:

١- انشاء موقع للمشروع وحفظه على جهاز الكمبيوتر داخل المجلد htdocs

٢- انشاء صفحات الموقع

ولإنشاء موقع من خلال تطبيق Expression Web ، قم تحت اشراف معلمك بتنفيذ خطوات نشاط إنشاء موقع باستخدام تطبيق Expression Web الموضحة فى كتاب الأنشطة والتدريبات

تـــــــــذكر الصـــــــــفحات التـــــــي ســـــــيتم إنشـــــائها فـــــــي الموقـــــــع المقتـــــــرح،

صفحات الموقع، هي:

- صفحة إدخال مصطلح "Add_term.php" والتي تستخدم في إضافة مصطلح جديد.
- صفحة البحث عن مصطلح "Search_term.php" وتستخدم في البحث عن مصطلح وعرض بياناته.
- صفحة تعديل بيانات مصطلح "Edit_term.php" ومن خلالها يمكن تعديل أي بيان من بيانات مصطلح معين.
- صفحة حذف مصطلح "Del_term.php" ومن خلالها يمكن حذف مصطلح معين.
- صفحة المساعدة "Help.php" ومعلومات حول كيفية التعامل مع موقع "قاموس مصطلحات الكمبيوتر المصور.
- صفحة الاتصال بقاعدة "Connection.php".
- **صفحة "Header.php" : تحتوى الصفحة على الصورة التي تظهر أعلى صفحات الموقع جميعها ، وتحتوى الروابط بجميع صفحات الموقع، والجزء المخصص للتحقق من تسجيل دخول المستخدم على الموقع أم لا.**
- **الصفحة الرئيسة ""Index.php:** واجهة استخدام الموقع ، وتحتوي الروابط التي من خلالها يمكن الانتقال إلى باقي صفحات الموقع .

مما سبق نلاحظ:

- تم فصل كود الربط بقاعدة البيانات في صفحة Connection.php.
- تم فصل جميع البيانات المشتركة في جميع صفحات الموقع في صفحة باسم "Header.php".
- يمكن التجول/ الإبحار عبر صفحات الموقع من خلال الصفحة الرئيسة للموقع **Index.php**.

نظرًا لخطورة عمليتي التعديل في بيانات المصطلح أو حذفه من الموقع ، فقد تم منح هذه الصلاحية للمستخدم الذي يقوم بتسجيل دخوله فقط، وهو ما ستقوم بتنفيذه فى مرحلة تأمين الموقع في الوحدة الرابعة من هذا الكتاب، وبذلك ينبغي على المستخدم الذي سجل دخول أن يقوم بتسجيل خروج من الموقع، وذلك بعد انتهاء تفاعله مع موقع "قاموس مصطلحات الكمبيوتر المصور".

يعتمد إنشاء صفحات الموقع على استخدام لغة الترميز HTML ، لذا نتطرق الآن إلى أهم المعلومات والمهارات الأساسية في لغة الترميز HTML وتطبيق **Expression Web**، للمساعدة على إنشاء صفحات الموقع

لغة الترميز HTML "Hyper Text Markup Language"

لغة تكويد تستخدم في إنشاء صفحة ويب ثابتة تحفظ بامتداد **.htm** أو **.html** بحيث يمكن عرضها وترجمة الكود الخاص بها من خلال أحد برامج مستعرضات الإنترنت.

يمكن لكتابة كود **HTML** باستخدام أحد برامج معالجة النصوص مثل: **MS Word, WordPad, Notpad and** ....

مع مراعاة الآتي:

- الأمر الذي نكتبه بلغة الترميز **HTML** يطلق عليها **Tag**، ولا يشترط أن يكتب بحروف كبيرة **Capital** أو صغيرة **Small**.
- أن يكتب الكود وفق بنية محددة



شكل (١١) يوضح الهيكل البنائي لبرنامج بلغة الترميز HTML

خصائص لغة الترميز HTML

(١) الأوامر **Tags** في لغة الترميز **HTML** توضع بين العلامتين: **>** **<**.

(٢) معظم الـ **Tags** لها بداية أو فتح **<.....>** ولها نهاية أو إغلاق **</.....>** كما يلي:

<Tag> النص الذي سيؤثر عليه الأمر </Tag>

شكل (١٢) الشكل العام للـ Tag في لغة الترميز HTML

مع ملاحظة الآتي:

- لكل **Tag** عمل معين يقوم به.
- كل **Tag** يكتب من اليسار إلى اليمين.
- **<Tag>** بداية الـ **Tag**.
- **</Tag>** نهاية الـ **Tag**.
- بعض **Tags** ليس لها إغلاق أو نهاية.

- بعض الـ **Tags** لها خصائص يمكن تعيينها معه في بداية الـ **Tag**.

(٣) يبدا الكود بــ **<Html>** وينتهي بـ **</Html>** وبينهما الآتي:

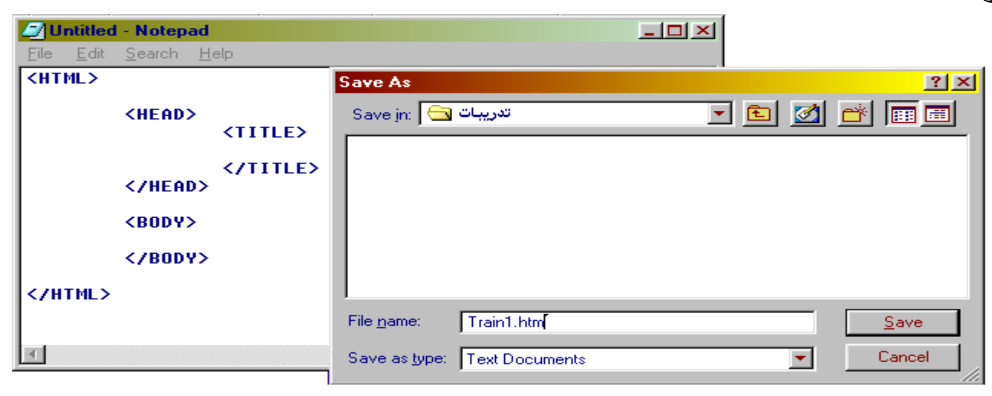
</Head> <Head>	يستخدم في عرض معلومات مثل: وصف للموقع وعن كاتب الكود واللغة ...إلخ.
</Title> <Title>	يختص بكتابة عنوان لصفحة الويب يظهر في **Title Bar** لشاشة مستعرض الإنترنت وهو جزء من **<Head>**.
</Body> <Body>	نضع فيه كافة أنواع المعلومات المراد عرضها على صفحة مستعرض الإنترنت من النصوص والتأثيرات التي تتم عليها والصور والفيديو والأصوات والجداول والارتباطات ... إلخ.

نهاية الـ Tag		بداية الـTag
</ HTML>	جميع محتويات الصفحة	<HTML>
</ HEAD>	رأس الصفحة	<HEAD>
</ TITLE>	عنوان الصفحة	<TITLE>
</ BODY>	محتوى الصفحة	<BODY>

شكل (١٣) ملخص الأقسام الرئيسة لكود HTML

يشترط لحفظ كود HTML في ملف مراعاة أن:

- يحفظ ملف كود HTML بأي اسم يكون امتداده **.htm** أو **.html** .
- نوع الملف **.txt**، ويفضل استخدام برنامج المفكرة **Notpad** لأنه يحفظ الملف بهذا النوع مباشرة دون اختيار منا.



شكل (١٤) حفظ هيكل برنامج HTML المكتوب في برنامج المفكرة (Notepad)

- بعد حفظ الملف بالمواصفات السابقة يأخذ الملف رمز مستعرض الإنترنت المتاح على جهازك مثل: Internet Explorer أو Google Chrome أو Mozilla Firefox ... الخ.
  يكون رمز الملف هو في حالة internet explorer.

نشاط ٢: "إنشاء صفحة ويب" باستخدام كود HTML

تحت اشراف معلمك اتبع إجراءات النشاط بكتاب الأنشطة و التدريبات ص لاستخدام كود HTML في إنشاء صفحة ويب.

ناقش مع معلمك الجدول التالي الذى يحتوي ملخصًا للغرض من بعض أكواد HTML المستخدم في النشاط:

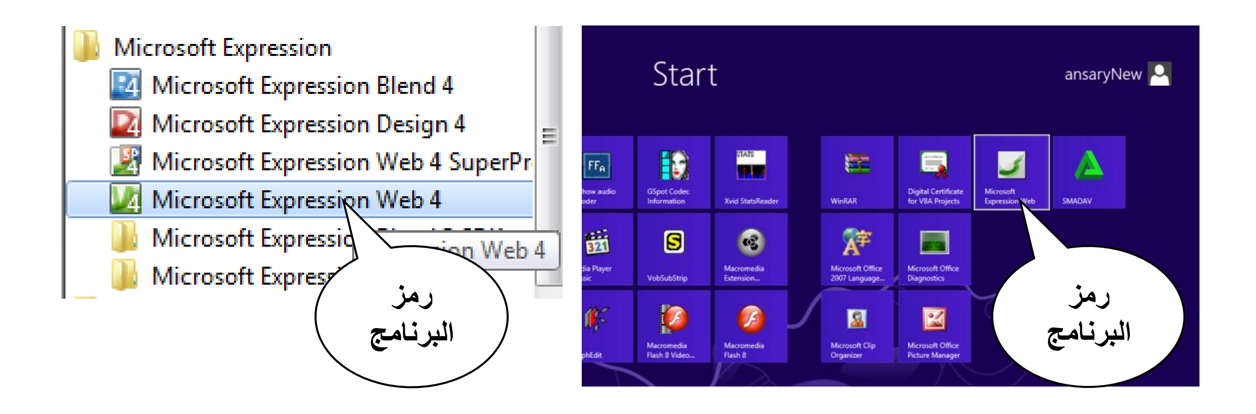
الشرح	الكود
إضافة عنوان لصفحة مستعرض الإنترنت.	<Title> My First Web Page </Title>
كل نص يكتب في <Body> يظهر على صفحة المستعرض، ناحية اليسار باعتباره الاتجاه الافتراضي للكتابة على صفحة المستعرض.	<Body> الكمبيوتر التعليمي
الكود <P> فقرة جديدة، و"Center" = align خاصية تعني محاذاة النص التالي لها توسيط على شاشة المستعرض إلى أن يُغلق الـ Tag كما يلي </P>.	<p align = "Center">
img src تعني "Image Source" مصدر الصورة ويكتب بعد علامة "=" حيث يشير إلى المسار Path واسم ملف الصورة بالامتداد.	<img src = "Egypt.jpg">
H تعني "Head" يقصد بها أن النص الذي يليها عنوان ويظهر بحجم خط حسب الرقم الذي يلي حرف H ويكون من ١ إلى ٦ حيث يشير الرقم ١ إلى أكبر خط والرقم ٥ خط أقل حجمًا وهكذا، وتظهر عبارة "تحيا مصر" أيضًا في منتصف السطر حيث أن <P> لم يُغلق بعد.	<H1> تحيا مصر </H1>
كتابة نص على صفحة المستعرض محاذة يمين.	<p align = "right"> الصفحة الثانية </p>

يمكنك عزيزى الطالب انشاء صفحة ويب باستخدام العديد من التطبيقات أو البرامج الجاهزة بسرعة وسهولة بدًلا من كتابة أكواد HTML، واليك الطريقتين :

وفيما يلى شرحًا مختصرًا لخطوات تحميلك تطبيق Expression Web على جهازك، لتوظيفه فى الربط بين صفحات الويب (نشاط ٣)

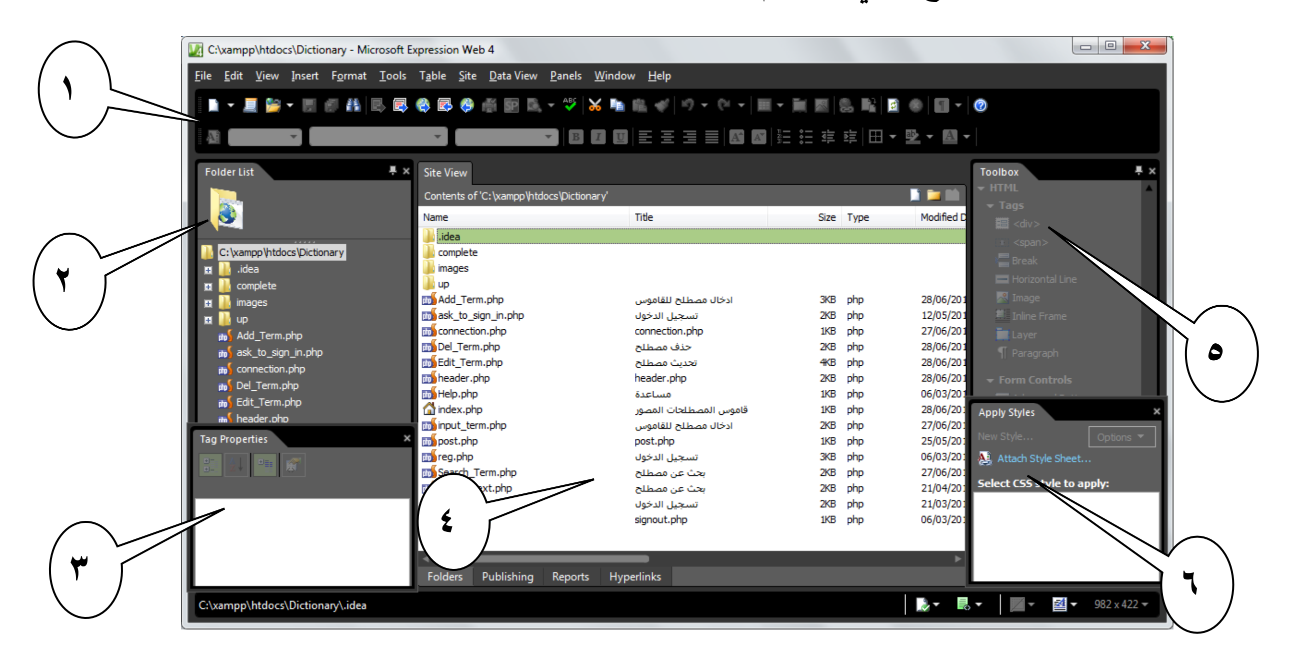
أوّلًا: من خلال إحدى تطبيقات إنشاء مواقع ويب

يمكن تحميل برنامج **Expression Web** لإنشاء مواقع وصفحات ويب من خلال واجهة تطبيق سهلة وبسيطة بدًلا من كتابة أكواد **HTML**، من نافذة **Start**:



شكل (١٥) تحمل تطبيق Expression Web

لتظهر الشاشة الافتتاحية للبرنامج والتي من أهم مكوناتها:



شكل (١٦) مكونات الشاشة الرئيسة لبرنامج Expression Web

(١) شريط القوائم **Menu Bar**.

(٢) نافذة قائمة المجلدات **Folder List**.

(٣) نافذة الخصائص **Tag Properties**.

(٤) مكونات موقع الويب **Web Site**.

(٥) نافذة صندوق الأدوات **Toolbox**.

(٦) نافذة أنماط التطبيق **Apply Style**.

وتتميز التطبيقات التي تستخدم في إنشاء مواقع وصفحات ويب بخاصية **(WYSIWYG)** "**What You See Is What You Get**" بمعنى: أن ما تراه هو ما تحصل عليه، كما يمكن عرض كود **HTML** الخاص بصفحة الويب والتعديل فيه، فمثلا يمكننا استخدام التطبيق في الربط بين صفحتي الويب من خلال شريط الأدوات وذلك بعمل ارتباط تشعبي **Hyperlink**، على نص أو صورة بديلًا لكود **HTML** التالي:

```
<a href = "العنوان المراد الانتقال إليه">  النص أو الصورة المراد ربطها بالعنوان  </a>
```

ثانيًا: في كود لغة الترميز HTML.

عزيزي الطالب والطالبة، بعد إنشاءك صفحتي ويب باستخدام لغة الترميز HTML، تحتاج للتعديل في صفحتي الويب والربط بينهما من خلال نص أو صورة، وهو ما يمكنك تنفيذه بتعديل الكود في لغة الترميز HTML

تحت اشراف معلمك قم بتنفيذ نشاط ٤ الذى يساعدك على الربط بين صفحات الويب في كود لغة الترميز HTML

نشاط ٤ : الربط بين صفحات الويب في كود لغة الترميز HTML
كتاب الأنشطة والتدريبات ص

إنشاء صفحة "Header.php"



شكل (١٧) صفحة "Header.php"

ادرس الشكل السابق ، وحدد الأجزاء المتكررة في جميع صفحات الموقع

تحتوى هذه الصفحة على الأجزاء المتكررة في جميع صفحات الموقع كما –يتضح من الصورة السابقة– مثل:

- الصورة في بداية كل صفحة ويطلق عليها **Banner**.
- النصوص التالية: الرئيسة – إدخال مصطلح – بحث عن مصطلح – تعديل – حذف مصطلح – مساعدة.

ملاحظة هامة: النصوص السابقة تظهر على صفحة المستعرض مجرد كلمات غير مرتبطة بصفحات الموقع أي أنها لم تصبح بعد روابط تشعبية HYPERLINK.

وصفحة "**Header.htm**" تهدف باختصار إلى إدراج صورة ثم كتابة بعض النصوص التي تفصل بينها مسافات، كما يتضح من شاشة الكود التالية:

```
Site View   header.php*
1  <html  dir="rtl">
2      <head>
3      <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
4      </head>
5      <body>
6              <div>
7              <img  height='161' src='images/banner.jpg' width='1267' >
8              </div>
9          <div  style="width: 1256px;text-align:right"    >
10          
11         الرئيسية
12               
13         إضافة مصطلح
14                
15         البحث عن مصطلح
16                
17         حذف مصطلح
18                
19         تعديل مصطلح
20                
21         مساعدة
22         </div>
23     </body>
24 </html>
Design   Split   Code
```

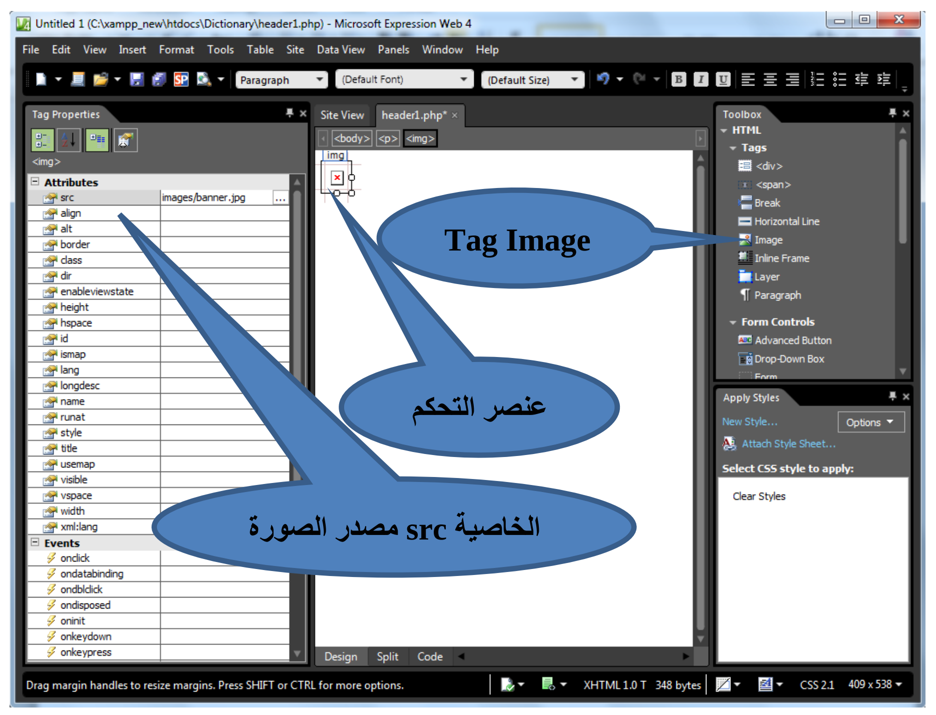
شكل (١٨) نافذة كود صفحة "Header.php"

نلاحظ في الكود الذي تم إنشاؤه ما يلي:

- يستخدم الـ **Tag** التالي لإنشاء مقطع بالصفحة:

```
<div>
كود HTML الخاص بهذا المقطع
</div>
```

– **المقطع الأول:** خاص بإدراج صورة **Banner** من الملف **'images/banner.jpg'**، مع ضبط خاصيتي الارتفاع والعرض.



شكل (١٩) إنشاء صور Banner وضبط خصائصها

```
<div>
<img  height='161' src='images/banner.jpg' width='1267' />
</div>
```

– **المقطع الثاني:** تم ضبط بعض خصائصه من عرض ومحاذاة يمين، ثم كتابة اسماء صفحات الموقع، وترك بعض المسافات بالكود ** ** وهو اختصار يعني "**Non Breakable Space**" ويفيد في ترك مسافة واحدة كما يظهر بالكود التالي:

```
9           <div  style="width: 1256px;text-align:right"    >
10           
11          الرئيسية
12                
13          إضافة مصطلح
14                 
15          البحث عن مصطلح
16                 
17          حذف مصطلح
18                 
19          تعديل مصطلح
20                 
21          مساعدة
22          </div>
```

شكل (٢٠) جزء الكود الخاص بعناوين صفحات الموقع

كما يمكن إنشاء صفحة "**Header.php**" بلغة الترميز **HTML** أو أحد تطبيقات إنشاء صفحات ويب بالكود التالي:

```
1  <html  dir="rtl">
2  <head>
3  <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
4  </head>
5      <body>
6        <div>
7        <img  height='161' src='images/banner.jpg' width='1267' >
8        </div>
9
10         <div  style="width: 1256px;text-align:right"    >
11         <a href="index.php">الرئيسية</a>
12               
13         <a href="add_term.php"> إضافة مصطلح
14         </a>       
15         <a href="search_text.php">البحث عن مصطلح</a>
16                
17         <a href="Del_Term.php">حذف مصطلح</a>
18                
19         <a href="Edit_Term.php">تعديل مصطلح
20         </a>       
21         <a href="Help.php">مساعدة </a>
22         </div>
23     </body>
24 </html>
```

شكل (٢١) كود إنشاء صفحة Header.php

ونلاحظ في كود HTML السابق ما يلي:

(١) تم فتح مقطع **<div>** وإغلاقه **</div>** ووضع بينهما كود إدراج الصورة التي نرغب أن تكون **Banner** يظهر في بداية جميع صفحات الموقع.

(٢) المقطع الثاني يؤدي إلى عمل تنسيق محاذاة يمين لكل ما يليه، وحتى إغلاق المقطع، مع جعل النصوص العربية ارتباطات **Hyperlink** بصفحات الموقع.

(٣) الكود التالي:

```
<a href = 'index.php'> الرئيسة </a>
```

يشير إلى:

**** بداية كود إدراج ارتباط تشعبي بالعنوان وهو هنا "اسم الصفحة" الذي يلي علامة "=" والذي قد يكون اسم ملف أو اسم برنامج أو اسم صفحة ويب أو عنوان إنترنت ...إلخ، ثم إغلاق الكود بـ ****.

(٤) ** ** اختصار يعني "**Non Breakable Space**" وتفيد في ترك مسافة واحدة ولترك ست مسافات بعد كلمة "الرئيسة" تم كتابتها ٦ مرات يلي كلٍ منها ";".

(٥) وبنفس الفكرة تستطيع عزيزي الطالب أن تكرر الكود السابق مع استبدال كلمة "رئيسة" بنص "إدخال مصطلح"، ثم نص "بحث عن"، ثم نص "تعديل مصطلح ثم نص حذف مصطلح"، ثم كلمة "مساعدة" ليصبح كلٍ منها ارتباط تشعبي.

والصفحة "**Header.php**"، يتم استدعاؤها بأمر واحد بسيط في بداية كل صفحة من صفحات الموقع بدلًا من تكرار الكود.

باستخدام برنامج EXPRESSION WEB يمكنك إنشاء صفحة HEADER بإدراج صورة، وكتابة بعض النصوص تحتها، وإجراء بعض التنسيقات عليها، لتظهر بشكل مناسب. وهو ما ستقوم بتنفيذه تحت اشراف معلمك فى النشاط ٥ ""إنشاء صفحة HEADER". بكتاب الأنشطة والتدريبات ص

٢-٥ إنشاء صفحة الاتصال بقاعدة البيانات Connection.php

نحتاج في جميع صفحات موقع "قاموس مصطلحات الكمبيوتر المصور" إلى إضافة مصطلح أو تعديل بياناته أو البحث عنه أو حذف مصطلح من قاعدة البيانات، الأمر الذي يتطلب منا فتح اتصال بقاعدة البيانات حتى يتسنى لنا إجراء هذه العمليات على بيانات قاعدة البيانات.

عند إنشاء صفحة الاتصال بقواعد البيانات نحتاج إلى التعامل مع متغيرات، وهو ما سنلقي عليه بعض الضوء قبل إنشاء صفحة الاتصال بقواعد البيانات:

المتغيرات والثوابت في لغة PHP

لقد تعرضت عزيزي الطالب في الحلقة الإعدادية لمفهوم المتغيرات والثوابت، والمفهوم لا يختلف عن ما سبق دراسته، ولكن دعنا نلقِ بعض الضوء من منظور لغة **PHP**، فالمتغير عبارة عن مخزن في الذاكرة يحدد له اسم ونوع ويخصص له قيمة تخزن فيه، وتتغير قيمته أثناء سير البرنامج، فمثلًا إذا كان المتغير **$Total** يساوي ١٠٠، فإنه يمكن تغيير القيمة التي تم تخصيصها بقيمة أخرى في لغة **PHP**، بجملة التخصيص التالية:

```
$Total = 450;
```

نلاحظ الآتي:

- يتم الاعلان عن المتغير عند استخدامه.
- أن اسم المتغير يبدأ بعلامة "$".
- يتكون اسم المتغير من أحرف و أرقام وعلامة "_" فقط، مثل: **$user_name – $A123 – $Password .... .**
- يفضل أن يكون اسم المتغير معبرًا عن محتواه أو ما يشير إليه.
- علامة التخصيص هي "=".
- تنتهي كل جملة في لغة **PHP** بعلامة "**;**".

• لطباعة أي معلومات على صفحة المستعرض نستخدم **Print** أو **Echo** هكذا:

- **print $total;**
- **OR echo ("$total");**

كما أنه يمكن استخدام كود **Echo** لتجميع أكثر من ثابت أو متغير عند الطباعة على صفحة المستعرض وذلك بالفصل بينها بنقطة " . ".

وهناك صيغة أخرى لطباعة قيمة المتغير كما يتضح من المثال التالي:

```
$name="Mohamed";
echo "my name is $name";
echo 'my name is $name';
echo 'my name is  ' .$name;
```

عند تنفيذ الكود السابق تكون نتيجته على شاشة مستعرض الإنترنت كما يلي:

my name is Mohamed

my name is $name

my name is Mohamed

نلاحظ من تنفيذ الكود السابق ما يلي:

- في جملة الطباعة الأولى تم طباعة قيمة المتغير عند وضعها داخل علامتي التنصيص المزدوجة **Double quotation** " ".
- بينما تم اعتبار المتغير بين علامتي التنصيص المفردة **Single quotation** ' ' نص وطبع بالشكل التالي **my name is $name.**
- في جملة الطباعة الثالثة تم استخدام النقطة لطباعة سلسلة حرفية وقيمة المتغير.

• يمكن استعراض بعض أنواع المتغيرات في لغة **PHP** من خلال الجدول التالي:

Type	Example
String	”Mohammad“

Integer	123
Double	1.23
Boolean	True / False

- والجدير بالذكر أنه لا يتم الإعلان عن المتغير ونوعه في لغة **PHP** وإنما يتعرف مترجم **Apache** على نوع المتغير من خلال القيمة المخصصة للمتغير في جملة التخصيص.
- فمثلًا: تستطيع عزيزي الطالب معرفة نوع أي متغير باستخدام الدالة **gettype($var);** وذلك بالكود التالي:

```
<?php
   $U_name="mohamed";        // متغير يساوي قيمة حرفية
   echo gettype($U_name);        //  طباعة نوع المتغير
   echo "<br />";                // النزول على السطر التالي
   echo "Welcome to PHP"; // PHP طباعة رسالة ترحيب بلغة
   echo "<br />";                // النزول إلى السطر التالي
   echo gettype($test);          //  طباعة نوع متغير آخر لم يتم استخدامه
?>
```

– نلاحظ أن ناتج تنفيذ الكود طباعة نوع المتغير **$U_name** على صفحة مستعرض الإنترنت بأنه "**String**"، ثم يترك سطر ويطبع رسالة الترحيب "**Welcome to PHP**"، ثم يترك سطر ويطبع **Null** أي قيمة المتغير فارغ نظرًا لعدم تخصيص قيمه له.

أما الثوابت فهي مخازن في الذاكرة تحمل اسم ونوع ويخصص لها قيمة ثابتة لا تتغير طوال تنفيذ البرنامج، ويمكن تعريفها من خلال الصيغة العامة التالية:

```
define('Constant Name', Value);
```

حيث:

- **Constant Name تشير إلى اسم الثابت.**
- **Value تمثل القيمة المخصصة للثابت.**

مع ملاحظة أنه لا يتم وضع علامة $ أمام اسم الثابت عند استخدامه.

فعلى سبيل المثال:

```
define('name', 'Mohamed');
echo('my name is:  '. name);
```

- **تستخدم النقطة (.) في سطر الكود الثاني لربط سلسلتين حرفيتين.**
- **تم استخدام الثابت بدون علامة $.**
- **استخدمت كلمة define لتعريف الثابت.**
- **ينتهي كل سطر في لغة PHP بفاصلة منقوطة (;).**

عند تنفيذ الكود السابق تكون نتيجته على شاشة مستعرض الإنترنت كما يلي:

my name is: Mohamed

والجـــدير بالـــذكر أن المتغيـــرات والثوابـــت تتـــأثر بحالـــة الأحـــرف كبيـــرة أو صـــغيرة (**Case-insensitive**)، وإذا أردت أن يكون الثابت غير حساس لحالة الأحرف نستخدم الصيغة التالية:

```
define('Constant Name', Value, true);
```

ومن الضروري التأكيد على الآتي:

(١) أن كود **PHP** يكتب وفق الصيغة التالية:

```
<?php
                    كود PHP المراد تنفيذه
?>
```

(٢) الكود **php?>** يعلن أن الكود التالي عبارة عن **Script** بلغة **PHP** سيتم تنفيذه على جهاز الخادم من خلال **Apatche Server**.

(٣) كود الطباعة على صفحة المستعرض في لغة **PHP** هو:

```
Echo ("
        المحتوى المراد طباعته على صفحة المستعرض أو أي كود بلغة HTML
    ");
```

(٤) كل جملة في لغة **PHP** لابد أن تنتهي بفاصلة منقوطةSemi Colon ";".

(٥) كتابة تعليق داخل كود **PHP** دون أن يترجم أو ينفذ يسبق بعلامة "//" أو، أما لكتابة ملاحظة أو نص أكثر من سطر فيتم كتابة التعليق أو الملاحظات بالكامل مسبوقة بعلامة "*/" وفي نهاية الملاحظة نضع العلامة "**/***".

بعد إنشاء قاعدة البيانات **dbdictionary** التي تحتوي على مصطلحات الكمبيوتر في جدول **Terms**، سوف نتعامل مع سجلات جدول البيانات من خلال صفحات موقع "قاموس مصطلحات المصور" حيث يمكننا من خلال صفحاته المختلفة إضافة مصطلح إلى قاعدة البيانات أو البحث عن مصطلح أو تعديل أي بيان فيه أو حذف مصطلح، وجميع هذه العمليات بداية تحتاج إلى إنشاء وفتح اتصال بين قاعدة البيانات وصفحات الموقع، وذلك قبل إجراء أيٍ من هذه المعاملات، ومن هنا سوف نبدأ في إنشاء صفحة تحتوي على كود **PHP** يحقق الاتصال بقاعدة البيانات بحيث يتم استدعاؤها في بداية كل صفحة ويب تقوم بإحدى عمليات التحديث في قاعدة البيانات.

Site View | connection.php*

```
1  <?php
2
3      $username="root";
4
5      $password="";
6
7      $database="dbdictionary";
8
9      $server="localhost";
10
11     $connect=mysql_connect("$server","$username","$password");
12     if ($connect)
13          {
14          $select=mysql_select_db("$database") or die("هناك مشكلة في قاعدة البيانات");
15          }
16          else
17          {
18          echo("لم يتم الاتصال بقاعدة البيانات");
19          }
20
21 ?>
```

شكل (٢٢) كود صفحة الاتصال بقاعدة البيانات

<?php

– هناك أربع متطلبات أساسية للاتصال بقاعدة البيانات هي: اسم المستخدم – كلمة المرور – اسم قاعدة البيانات – اسم جهاز الخادم.

```
$username="root";
$password="";
```

اسم مستخدم **mysql** يخصص للمتغير **$username**

كلمة المرور الافتراضية "" تخصص للمتغير **$password** ويجب تغييرها كما سندرس في الوحدة الرابعة.

```
$database="dbdictionary";
$server="localhost";
```

اسم قاعدة البيانات يخصص للمتغير **$database**

اسم الخادم "**localhost**" المُضيف لقاعدة البيانات يخصص للمتغير **$server**

```
$connect=mysql_connect("$server","$username","$password";
```

الغرض من الأمر السابق تخصيص ناتج تنفيذ دالة **mysql_connect** والتي تحتاج إلى المعطيات الثلاثة: (اسم جهاز الخادم المضيف، واسم المستخدم، وكلمة المرور)، في متغير الربط **$connect**، حيث يكون ناتج الدالة "**True**" إذا تم التحقق من صحة المعطيات الثلاثة السابقة، أو "**False**" إذا كانت غير ذلك، وللتأكد من ذلك نستخدم جملة **IF** التالية.

```
if ($connect)
```

```
{
    $select=mysql_select_db("$database") or die
     ("هناك مشكلة في قاعدة البيانات");
}
else
{
  Echo ("لم يتم الاتصال بقاعدة البيانات");
}
```

الشرط يسأل عن قيمة المتغير **$connect**، فإذا كانت "**True**" فإنه يتم الاتصال بقاعدة البيانات المسماة **$Database** من خلال الدالة **mysql_select_db** أو عرض رسالة حال وجود مشكلة، من خلال الكود التالي:

```
Or die ("هناك مشكلة في قاعدة البيانات");
```

وإذا كانت قيمة المتغير **$connect**، "**False**" يعرض الرسالة التي تلي **else** على صفحة المستعرض بأنه "**لم يتم الاتصال بقاعدة البيانات**".

```
?>
```

والآن قم بإنشاء صفحة الاتصال بقواعد البيانات "connect.php" التدريب( ) بكتاب الأنشطة والتدريبات

الوحدة الثالثة

إنشاء صفحات موقع "قاموس مصطلحات الكمبيوتر المصور"

فى نهاية الوحدة يتوقع أن يكون الطالب قادرًا على أن:

١-يتعرف أدوات التعلم الإلكتروني المعتمد على الانترنت

٢-يوظف تطبيقات تكنولوجيا المعلومات والأتصالات في بناء محتوى تعلمه

٣-يوظف التطبقات المناسبة في معالجة عناصر الوسائط المتعددة (صوت -فيديو-صورة ...).

٤-يطور المهام الإلكترونية التي يمكنه تنفيذها لدعم تعلمه.

٥-يستخدم تكنولوجيا المعلومات والاتصالات فى التواصل مع الخبراء فى المجالات المختلفة

٦-يمارس مهارات التعامل مع المعلومات الإلكترونية (تحديد – تقييم – انتقاء – تنظيم).

٧-يوظف المتعلم مصادر المعلومات الإلكترونية والمرتبطة بمشروع محدد.

الموضوع الأول

المعاملات والجمل الشرطية فى لغةPHP

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يميز بين المعاملات Operator فى لغة PHP.

٢- يشرح جملة الشرط If condition فى لغة PHP .

٣- يتعرف جملة Switch فى لغة PHP.

٤- يمارس الصيغ المختلفة لجملتى الشرط Switch& If .

٥- ينشيء صفحة Header باستخدام تطبيق Expression Web.

قبل البدء في إنشاء صفحة إضافة مصطلح يمكن أن نستعرض معًا مفهوم المعاملات **Operator**، حيث يوجد ثلاثة أنواع في لغة PHP وهي:

(١) **معاملات حسابية بنوعيها (العادية – أخرى).**

المعامل	يشير إلى عملية	مثال	ناتج المثال
+	الجمع	2 + 3	5
–	الطرح	6 – 4	2
*	الضرب	5 * 2	10
/	القسمة	8 / 2	4
%	باقي القسمة	10 % 3	1

وهناك معاملات حسابية أخرى خاصة بلغة PHP

Operation	Example	Means
+ =	$x +=5	$x = $x + 5
– =	$x –=5	$x = $x – 5
* =	$x *= 5	$x = $x * 5
/ =	$x /= 5	$x = $x / 5
% =	$x % = 5	$x = $x % 5
+ +	$x++	$x = $x + 1
- -	$x–	$x = $x – 1

(٢) معاملات مقارنة

Operator	==	!==	>	>=	<	<=
Refer to	Equal	Not Equal	Greater Than	Greater Than Or Equal	Less Than	Less Than Or Equal

(٣) معاملات منطيقة Logical Operator

Operator	Name	Result True	Example	Result
||	أو	أحد الطرفين صحيح	True || False	True
OR	أو	أحد الطرفين صحيح	True OR False	True
&&	و	كلا الطرفين صحيح	False && True	False
AND	و	كلا الطرفين صحيح	True AND False	False
!	لا	الطرف ليس صحيح	True!	False

جملة الشرط IF Condition

جملة الشرط IF، والتي يمكن من خلالها لمطوري البرامج وضع شرط معين واختباره فإذا كان ناتج الشرط صحيحًا يقوم البرنامج بتنفيذ كود معين، وإذا كانت نتيجة تنفيذ الشرط غير صحيحة قام بتنفيذ كود آخر.

ويمكن استخدام جملة **IF** بصور متعددة حسب متطلبات البرنامج، ولعل أبسط صورة تستخدم فيها جملة IF في لغة **PHP** هي:

```
If (Logical Condition)
    {
        الكود المراد تنفيذه إذا كان ناتج الشرط صحيحًا
    }
```

حيث:

Logical Condition تشير إلى تعبير شرطي مثل: **50 < Degree** أو **6 > Age** أو "**Administrator**" == **Name** ...إلخ.

فكما يتضح من الأمثلة وما سبق دراسته نجد أن الشرط يقارن بين متغيرين أو متغير وثابت بينهما إحدى معاملات المقارنة السابق الإشارة إليها.

وفي هذه الحالة يتم فقط تنفيذ الكود بين القوسين { } إذا تحقق الشرط.

مثال:

```
<?PHP
    $A="First";
```

```
        If ($A=="First")
            {
               Echo "أنت الأول";
            }
?>
```

نلاحظ في المثال السابق: أن عبارة "أنت الأول" ستطبع فقط على صفحة مستعرض الإنترنت إذا كانت فقط قيمة المتغير **$A** تساوي "**First**"، كما نلاحظ أن علامة "=" في الكود (**$A="First";**) تمثل معامل تخصيص، أما علامة "==" مع جملة **If** تمثل معامل منطقي، وأخيرًا تذكر أن كل جملة في لغة **PHP** تنتهي بفاصلة منقوطة "**;**".

والصورة التالية لجملة IF هي الأكثر شيوعًا

```
 If (Logical Condition)
        {
                 الكود المراد تنفيذه إذا تحقق الشرط True
        }
Else
        {
                 الكود المراد تنفيذه إذا لم يتحقق الشرط
        }
```

إذًا فالطبيعي أن أي شرط قد تكون نتيجته "**True**" فيتم تنفيذ الكود الخاص بتحقق الشرط فقط بعد **IF** مباشرة، وقد يكون نتيجة الشرط "**False**" فيتم تنفيذ الكود التالي لـ **Else** فقط.

مثال:

```
<?PHP
        $A="Second";
        If ($A=="First")
            {
               Echo "أنت الأول";
            }
      Else
            {
               Echo "أنت الثاني";
            }
?>
```

نلاحظ في المثال السابق: أن عبارة "أنت الأول" ستطبع فقط على صفحة مستعرض الإنترنت إذا كانت فقط قيمة المتغير **$A** تساوي "**First**"، وإلا سيتم طباعة عبارة "أنت الثاني"، ونظرًا لأن "**Second**" **=$A** فإن الشرط لم يتحقق، بالتالي يكون ناتج تنفيذ الكود بالكامل طباعة عبارة "أنت الثاني"، كما نلاحظ أن علامة "=" في الكود (**$A="First";**) تمثل معامل تخصيص، أما علامة "==" مع جملة **If** تمثل معامل منطقي، وأخيرًا لاحظ أننا لم نضع "**;**" داخل كود **If** بالكامل باعتبارها جملة واحدة.

وهناك الصورة التالية الأكثر تعقيدًا لجملة IF

```
If (Logical Condition)
        {
                الكود المراد تنفيذه إذا تحقق الشرط الأول
        }
Elseif (Logical Condition)
          {
                الكود المراد تنفيذه إذا تحقق الشرط الثاني
          }
      Else
         {
                الكود المراد تنفيذه إذا لم يتحقق الشرط الثاني
         }
```

نلاحظ هنا أن جملة **IF** الأولى يليها شرط إذا تحقق يتم تنفيذ الكود التالي لها مباشرةً وإذا لم يتحقق فهناك شرط ثان إذا تحقق ينفذ الكود الذي يلي **Elseif**، وإذا لم يتحقق ينفذ الكود الذي يلي **Else**، وبدءً من **Elseif** حتى نهاية الكود عبارة عن جواب عدم تحقق الشرط لجملة **IF** الأولى.

مثال:

```
<?PHP
          $X=5;
         If ($X<0)
             {
               Echo "الرقم سالب"
             }
         Elseif ($X>0)
                {
                  Echo "الرقم موجب"
                }
               Else
                  {
```

```
                    Echo "الرقم يساوي صفر"
                  }
?>
```

نظرًا لأن **$X=5** إذا فقد تحقق الشرط الثاني بعد **Elseif** وسيتم طباعة عبارة "الرقم موجب" على صفحة مستعرض الإنترنت.

وهناك صيغة أخرى لجملة IF

(Expression) ? If–True : If–False;

والمثال التالي يوضح استخدام الصيغة السابقة والتي يطلق عليها **In Line If:**

```
<?PHP
      $x = 1;                    // المتغير يساوي ١
      $y = ($x = = 1) ? 'One' : 'Two';
يتم اختبار قيمة المتغير $x إذا كانت قيمته تساوي ١، يحفظ القيمة "One" في المتغير $y وإلا يحفظ القيمة "Two" في المتغير $y.
      Print $y                   //  طباعة قيمة المتغير
?>
```

نجد في المثال السابق أن (:) **Colon** أدت إلى نفس النتيجة بدلا من استخدام الأقواس **{ }**.

ثانيا: جملة Switch

ويمكن أيضًا استخدام جملة **Switch** كأحد العبارات الشرطية في لغة **PHP**، حيث يمكن من خلالها اختبار عدة شروط وفي كل مرة يتحقق فيها شرط ما ينفذ كود معين، والمثال التالي يوضح ذلك:

```
1       <?php
2       $x=1;
3           switch ( $x )
4           {
5               case 1:
6                   echo "واحد";
7                   break;
8               case 2:
9                   echo "أثنين";
10                  break;
11
12              case 3:
13                  echo "ثلاثة";
14                  break;
15
16              default:
17                  echo "لايوجد";
18                  break;
19          }
20      ?>
21
22
```

شكل (٢٣) كود PHP يوضح استخدام الجملة الشرطية Swich

وبتحليل الكود السابق نجد أن:

المتغير **$x** بدأ بالقيمة ١، ثم استخدمت جملة **Switch** لاختبار قيمة المتغير **$x** حيث يوجد اكثر من احتمال:
فإذا كانت قيمة المتغير تساوي 1 أي "**case** 1:" يطبع "واحد"، وهكذا لباقي الاحتمالات وفي آخر احتمال إذا لم تكن قمة المتغير لا تساوي ١ أو ٢ أو ٣ أي خلافًا لذلك يطبع "لا يوجد".

وهناك أمر آخر يمكن الخروج به دون تنفيذ شيء هو أمر Exit، فمثلاً يمكن في آخر حالة Default نضع كلمة ;Exit للخروج من جملةٍ Switch بدلاً من طباعة "لا يوجد".

هناك صيغة أخرى لجملة **Switch** حيث يمكن أن تأخذ أكثر من قيمة واحدة كما بالمثال التالي:

```
<?php
    $a = 2;
    Switch ( $a )
    {
        case 1:

        case 2:

        case 3:

        echo 'واحد أو اثنين أو ثلاثة';
        break;

        default:
        echo 'أرقام أخرى';
    }
?>
```


شكل (٢٤) كود PHP يوضح صيغة أخرى لاستخدام الجملة الشرطية Swich

الموضوع الثاني

صفحة إدخال مصطلح Add_Term.php

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم واجهة صفحة إدخال مصطلح.

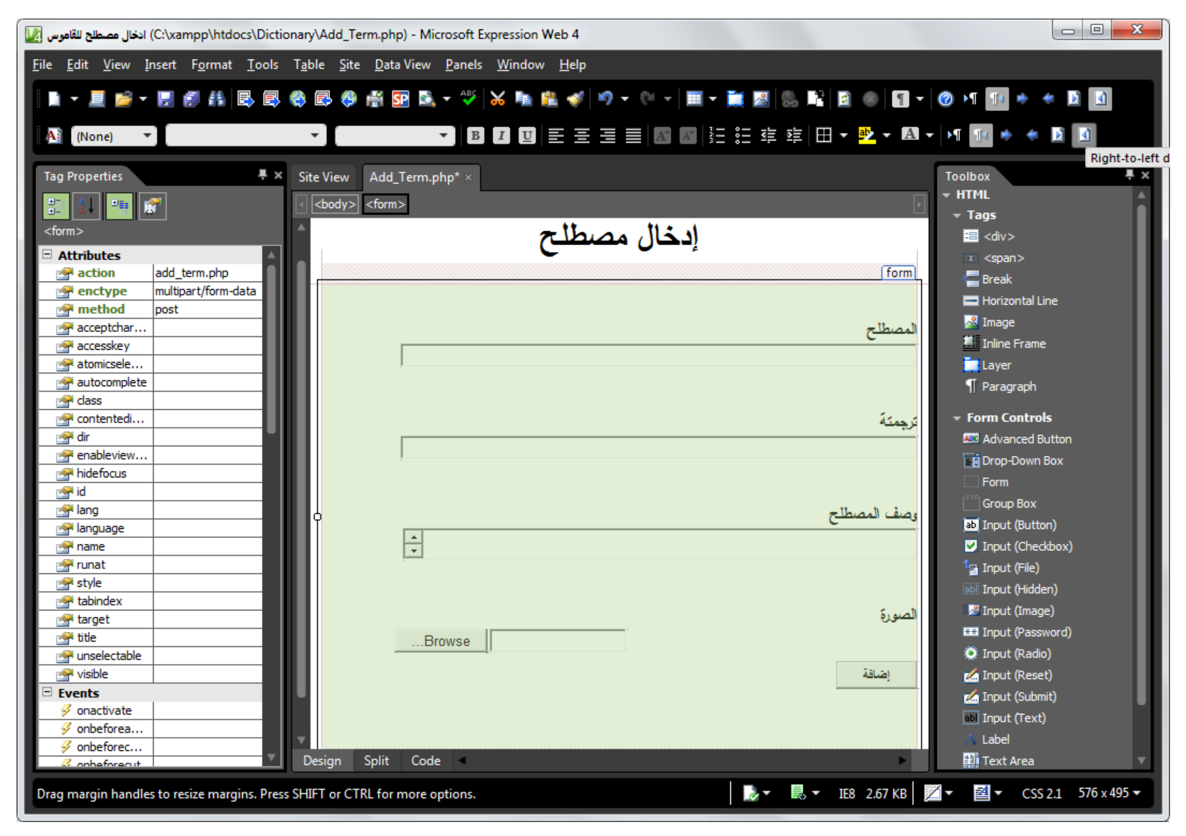
٢- يكتب كود PHP لتضمين صفحتي Header & Connection.

٣- ينشيء صفحة إدخال مصطلح.

٤- يتحقق من إضافة مصطلح إلى قاعدة البيانات.

(١) تصميم واجهة صفحة إدخال مصطلح

من خلال إحدى التطبيقات المستخدمة في إنشاء صفحات الويب يمكن إنشاء صفحة إدخال مصطلح كما بالشكل التالي:



شكل (٢٥) شاشة تصميم صفحة Add_Term.php

من الشكل السابق يتضح الآتي:

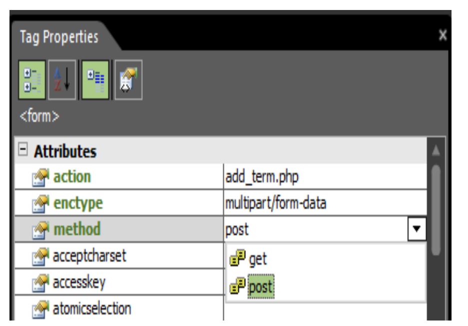
- كتابة عنوان شاشة الإدخال "إدخال مصطلح"، وقد تم وضع هذا العنوان في قسم مستقل <div>.
- وتستطيع عزيزي الطالب كتابة النصوص كما بالشاشة السابقة واستخدام نافذة الأدوات "Toolbox" أعلى يمين الشاشة لإدراج عناصر التحكم على الصفحة.

- **النمـوذج Form** يستخدم لتمرير أو إرسال كافة بيانات النموذج الموجودة في عناصر التحكم من مستعرض الويب إلى الخادم **Web Server** ويوجد طريقتين لإرسال بيانات النموذج وهما:

١- <**form method**="GET">

٢- <**form method**="POST">

حيث يمكن ضبط الخاصة **Method** للنموذج **Form** بالقيمة **POST** أو القيمة **GET**.



شكل (٢٦) ضبط خصائص النموذج Form

وإذا نظرت إلى كود **html** ستجد بداية الـ **Tag** الخاص بالنموذج **Form** كالآتي:

```
<form method="post" action="add_term.php" enctype="multipart/form-data">
```

من الصيغة السابقة نجد العديد من خصائص النموذج **Form** حيث:

- الخاصية **Method** يوجد لها معاملان إما المعامل **GET** أو المعامل **POST**.
- الخاصية Action ومعاملها اسم الصفحة التي يتم الانتقال إليها عند الضغط على زر **Submit**.

- بعض النصوص المُعبرة عن القيم المطلوب إدخالها في مربعات النصوص، مثل: نص "المصطلح" ويليه مربع نص لإدخال اسم المصطلح وهكذا باقي بيانات المصطلح، والصيغة التالية توضح كيفية إنشاء مربع نص على نافذة مستعرض الإنترنت:

```
<input name = "text_term" type = "text" style = "width : 482px" />
```

لاحظ فى الصيغة السابقة الآتي:

- **Input تعنى ادراج عنصر تحكم**
- **name = "text_term" تعنى اسمه text_term**
- **type = "text" تعنى نوعه مربع نص**
- **style = "width : 482px تعنى ضبط خاصية العرض لمربع النص**

- زر "إضافة" المسئول عن تنفيذ الكود الخاص بإدراج بيانات المصطلح الجديد في سجل بيانات في قاعدة البيانات "**db_dictionary**".
- تم وضع باقي مكونات الصفحة في قسم آخر **<div>**.

وللاطلاع على كود **HTML** المقابل لصفحة إدخال مصطلح الموضحة بالشاشة السابقة قم بتحديد النموذج **Form**، ثم اضغط زر "**Code**" اسفل نافذة البرنامج فيظهر الكود الخاص بهذا القسم محدداً كما بالشاشة التالية:

اضف كود php رقم ١

اضف كود php رقم ٢

```
1  <html  dir="rtl">
2      <head>
3      <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
4      <title>ادخال مصطلح للقاموس</title>
5      </head>
6      <body dir="rtl">
7
8       <div  style="text-align:center"  > <h1>إدخال مصطلح</h1></div>
9      <form method="post" action="add_term.php" enctype="multipart/form-data">
10         <div class="style2">
11             <br >   المصطلح             
12             <input name="txt_term" type="text" style="width: 482px" > <br >
13             <br >  ترجمة            
14             <input name="trans" type="text" style="width: 482px" > <br >
15             <br >  وصف المصطلح            
16             <textarea name="TextArea1" style="width: 480px; height: 30px"></textarea> <br/>
17             <br>  الصورة           
18             <input name="File1" style="width: 488px" type="file"> <br>    <br/>
19             <input name="Submit1" style="width: 76px" type="submit" value="إضافة" ><br> <br >
20             </div>
21     </form>
22
23         </body>
24 </html>
```

شكل (٢٧) كود HTML لصفحة Add_term.php

انتبه الى الكود التالي:



والذي يضم المعطيات الثلاثة التالية:

`<form method="post"` وهو أسلوب نقل البيانات من النموذج إلى الخادم Server.

`" action="add_term.php"` تحديد اسم الصفحة المراد الانتقال إليها عند الضغط على زر Submit.

`enctype="multipart/form-data">` تستخدم عندما يتم استخدام النموذج في رفع (الصور) على الخادم Server، وإذا لم يستخدم لا يتم رفع الصور داخل مجلد الصور "Pic".

(٢) كتابة كود PHP

عند هذا الحد نستطيع إدخال بيانات مصطلح في عناصر التحكم على صفحة المستعرض ولكنها بيانات غير مرتبطة بقاعدة البيانات، لذا نحتاج إلي:

- إظهار الصورة Banner وعناوين صفحات الموقع في بداية الشاشة وذلك من خلال إضافة كود PHP الخاص بإدراج صفحة Header في المكان المشار إليه بالرقم (١) على الشاشة السابقة بالكود التالي:

```
<?php
include("header.php");
?>
```

شكل (٢٨) كود PHP لتضمين صفحة Header

- تحقيق الاتصال بقاعدة البيانات.
- التحقق من إدخال بيانات في عناصر التحكم على صفحة المستعرض.

- تسجيل بيانات المصطلح في جدول قاعدة البيانات وذلك من خلال كود PHP المرتبط بزر "إضافة" في المكان المشار إليه بالرقم (٢) على الشاشة السابقة بالكود التالي:

```
47  <?php
48      include("connection.php");
49          if(isset($_POST['Submit1']))
50          {
51                  if(!is_dir('pic'))  { mkdir('pic'); }
52          @$file=$_FILES['File1']['name'] ;
53          @$tmp=$_FILES['File1']['tmp_name'];
54                  if(!empty($file))  {  move_uploaded_file($tmp,'pic/'.$file); }
55           $term=$_POST['txt_term'];
56           $trans=$_POST['trans'];
57           $defe=$_POST['TextArea1'];
58           $picture="pic/" . $file . " ";
59           if (  $term !== "" && $trans !== "" && $defe !== "" && !empty($file)  )
60             {
61             mysql_query("SET NAMES 'utf8'");
62             $query=mysql_query("insert into terms values('','$term','$trans','$defe','$picture')");
63                  if($query)
64                        {
65                        $datares="تم اضافة البيانات بنجاح";
66                        echo("<label id='Label1' style='color: #FF0000;font-size: x-large;'> $datares </label> ");
67                        }
68                  else
69                      {
70                        $datares="لم يتم اضافة البيانات";
71                        echo("<label id='Label1' style='color: #FF0000;font-size: x-large;'> $datares </label> ");
72                      }
73             }
74           else
75             {
76                 $datares="البيانات التى تم إدخالها غير صحيحة يجب إدخال جميع الحقول ";
77                 echo("<label id='Label1'  style='color: #FF0000;font-size: x-large;'> $datares </label>    ");
78             }
79        }
80  ?>
```

اضافة صفحة الاتصال بقاعدة البيانات

جملة شرطية لاختبار وجود مجلد **pic** وانشاؤه فى حالة عدم وجوده

جملة شرطية لاختبار هل تم اختيار ملف صورة

شكل (٢٩) كود PHP لإضافة مصطلح جديد

```
<?php
```

- تضمين صفحة تحقيق الاتصال بقاعدة البيانات بالكود التالي:

```
include("connection.php");
```

التحقق من الضغط على زر Submit

- الكود التالي يختبر إن كان المستخدم قد قام بالضغط Click على زر "إضافة" فيتم التحقق أوّلا من وجود مجلد باسم "pic" وهو المخصص لوضع الصور الخاصة بالمصطلحات بداخله، فإذا لم يُنشأ هذا المجلد يتم إنشائه.

```
if (isset($_POST['Submit1']))
 {
   if (!is_dir('pic')) { mkdir('pic'); }
```

- وبنفس الطريقة يمكن إنشاء مجلدات أخرى للصور أو الفيديو سواء بالكود أو من خلال نظام التشغيل في مجلد المشروع.

رفع ملف الصور على Server

- يخصص اسم الملف المختار من المستخدم File1 إلى المتغير $file، مع ملاحظة أن العلامة @ تسمى Pointer وتتعامل معها لغة Php على أنها متغير، ويفيد ذلك في عدم إظهار رسالة خطأ إذا كانت قيمته فارغة.

```
@$file=$_FILES['File1'] ['name'];
@$tmp=$_FILES['File1'] ['tmp_name'];
```

- اختبار إن كان اسم ملف الصورة غير محدد يتم تحميل ملف الصورة من مجلد "pic".

```
if (!empty($file))  {  move_uploaded_file($tmp,'up/'.$file);   }
```

تخصيص قيم عناصر التحكم في متغيرات

- يتم تخزين البيانات التي تم إدخالها في مربعات النصوص للمصطلح الجديد على صفحة المستعرض في المتغيرات التالية بالترتيب ($picture – $defe $trans – $term)، مع ملاحظة أن المتغير $_POST يخزن به قيم عناصر التحكم على النموذج Form عند الضغط على زر Submit.

```
$term=$_POST['txt_term'];
$trans=$_POST['trans'];
$defe=$_POST['TextArea1'];
$picture="pic/" . $file . " ";
```

التحقق من إدخال قيم في عناصر التحكم على النموذج Form

- جملة IF التالية للتأكد أن هذه المتغيرات ليست خالية أي تم إدخال قيم فيها، فإذا كان الشرط صحيحًا وجميع المتغيرات ليست خالية Null يقوم بتنفيذ الآتي:
- استخدام دالة mysql_query يساعد في إظهار البيانات على صفحة المستعرض باللغة العربية بشكل صحيح وعدم ظهورها في صورة علامات استفهام هكذا ؟؟؟؟؟؟؟.

```
if ($term !== "" && $trans !== "" && $defe !== "" &&
    !empty($file) )
  {
    mysql_query("SET NAMES 'utf8'");
    $query=mysql_query("into terms values ('','$term','$trans','$defe','$picture')");
```

variable متغير	Function دالة	Table الجدول	القيم المُدخلة	المتغيرات

– تم إنشاء متغير **$query** بحيث يخصص له قيمة تنفيذ الدالة **mysql_query** التي تحتوي على جملة **insert** الخاصة بلغة **SQL**، والتي تقوم بإدخال البيانات التي تم تخزينها في المتغيرات إلى الحقول المناظرة لها في جدول قاعدة البيانات "**terms**".

– ثم يتم التأكد من إدخال البيانات في جدول قاعدة البيانات **if($query)** فإذا تم ذلك ظهرت الرسالة "**تم إضافة البيانات بنجاح**"، وإلا تظهر الرسالة "**لم يتم إضافة البيانات**".

```
if ($query)
 {
   $datares="تـــــــــــم إضـــــــــــافة البيانـــــــــــات بنجـــــــــــاح";
   echo("<label                    id='Label1'                    style='color:
   #FF0000;font-size:         x-large;'>         $datares         </label>
   ");
 }
else
   {
    $datares="لم يتم إضافة البيانات";
    echo("<label id='Label1' style='color:
    #FF0000;font-size: x-large;'> $datares
    </label>       ");
   }
 }
else
```

– **في حالة عدم إدخال بيانات لجميع الحقول أو بعضها يطبع الرسالة "البيانات التي تم إدخالها غير صحيحة يجب إدخال جميع الحقول".**

```
 {
  $datares="البيانات التي تم إدخالها غير صحيحة يجب إدخال جميع الحقول";
  echo("<label id='Label1'  style='color:
  #FF0000;font-size: x-large;'> $datares </label>
  ");
 }
}
?>
```

عزيزى الطالب والطالبة، الآن يمكنك استخدام صفحة الويب كواجهة للتعامل مع جدول قاعدة البيانات لإضافة مصطلح جديد، وإجراء عمليات أخرى، وذلك من خلال تنفيذك نشاط ١: "إنشاء صفحة إدخال مصطلح Add_term.php" بكتاب الأنشطة والتدريبات

(٣) تنفيذ صفحة إدخال مصطلح

عند استعراض صفحة إدخال مصطلح على مستعرض الإنترنت تظهر كما يلي، وذلك من خلال العنوان localhost/dictionary/add_term.php:



شكل (٣٠) نافذة استعراض صفحة إدخال مصطلح

- في حالة عدم إدخال بيانات والضغط على زر "**إضافة**"، يطبع الرسالة "**البيانات التي تم إدخالها غير صحيحة يجب إدخال جميع الحقول**" أسفل شاشة المستعرض كما يلي:



شكل (٣١) نافذة استعراض صفحة إدخال مصطلح بشكل خاطيء

- عند إدخال جميع بيانات المصطلح كما يلي:



شكل (٣٢) نافذة إدخال مصطلح

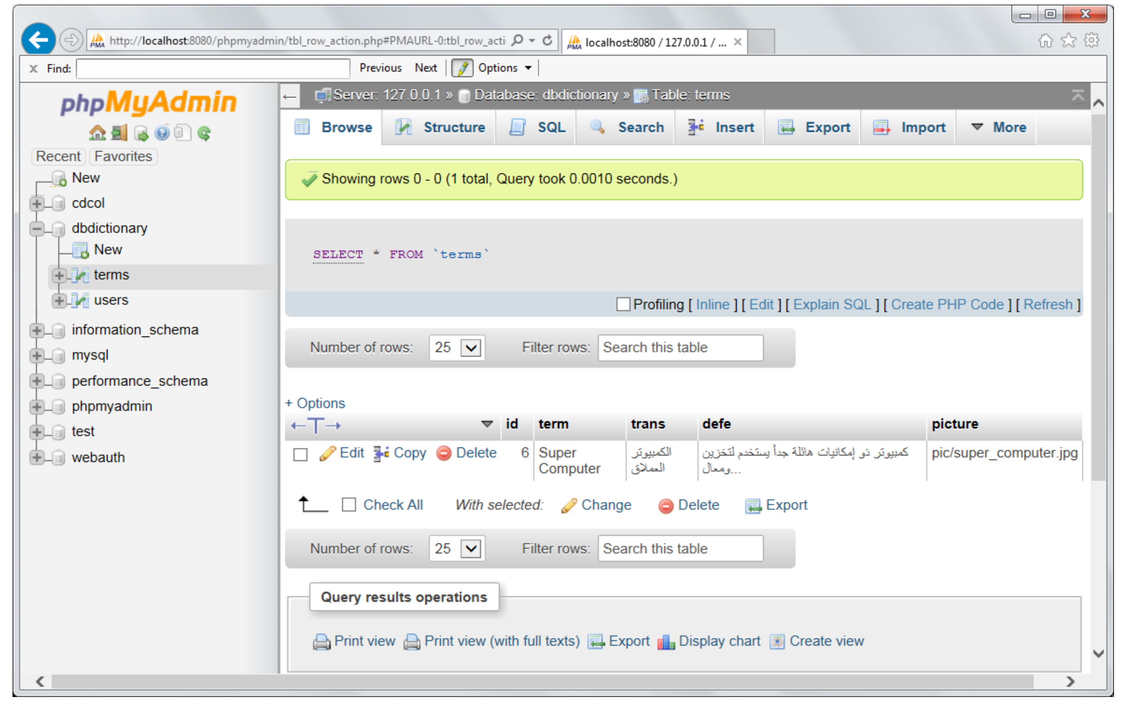
- وبالضغط على زر "**إضافة**" تظهر الرسالة "**تم إضافة البيانات بنجاح**" أسفل شاشة المستعرض.



شكل (٣٣) نافذة التأكد من إدخال مصطلح بشكل صحيح

(٤) التأكد من إضافة المصطلح إلى قاعدة البيانات

وللتأكد من أنه تم إضافة البيانات في سجل جديد في قاعدة البيانات، نفتح قاعدة البيانات في تبويب جديد لمستعرض الإنترنت بكتابة الرابط التالي "**localhost/phpmyadmin**" في شريط العنوان ونتأكد من إضافة سجل جديد في جدول "**terms**":



شكل (٣٤) نافذة التأكد من إضافة المصطلح في جدول قاعدة البيانات

تدريب

الموضوع الثالث

صفحة البحث عن مصطلح Search_term.php

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يشرح عبارات التكرار ....

٢- يصمم واجهة صفحة البحث عن مصطلح.

٣- يكتب كود **HTML** لإنشاء صفحة البحث عن مصطلح.

٤- يكتب كود **PHP**لتضمين صفحتى **Header & Connection**.

٥- يتعاون مع زملاءه فى كتابة كود **PHP** للبحث عن مصطلح.

٦- ينفذ عمليات البحث عن مصطلح.

قبل إنشاء صفحة البحث عن مصطلح ينبغي أن نلقِ الضوء على مفهوم التكرار **Looping** وبعض عبارات التكرار في لغة **PHP**، حيث نحتاج أحيانًا في بعض البرامج إلى تكرار كود معين عدة مرات أو تكراره لحين تحقق شرط معين، هنا نجد أن لغة **PHP** توفر لنا بعض عبارات التكرار مثل: **Do For – While** **While** وفيما يلي نضرب بعض الأمثلة التوضيحية لاستخدام عبارات التكرار السابقة في تطبيقات بسيطة:

(١) جملة While

استخدم جملة **While** في كتابة كود يطبع الأعداد من ١ : ١٠٠ بلغة **PHP**.

```
<?PHP
  $x = 1;                         // متغير يبدأ بالقيمة ١
  While ($x <= 100)               // تنفذ جميع الأوامر في جملة التكرار
                                  والتي تظهر بين القوسين { } طالما أن الشرط صحيح
                                  أي أن قيمة المتغير $x أقل من أو تساوي ١٠٠.
        {
         Echo ($x);               // طباعة قيمة المتغير
         Echo ("<Br>");           // النزول إلى السطر التالي
         $x ++;                   // زيادة العداد أو المتغير بقيمة ١
                                  ويمكن كتابة هذه الجملة بالصيغة $x = $x + 1;
        }
?>
```

نلاحظ من الكود السابق مدى الاختصار والبساطة في طباعة الأعداد من ١ : ١٠٠ وهو ما يوفر طباعة ١٠٠ جملة لطباعة الأعداد على شاشة المستعرض بالطريقة التقليدية.

مثال آخر: ما هو ناتج جمع العداد من ١ : ١٠٠ باستخدام لغة **PHP**.

```
<?php

    $x = 1;//بداية العداد
    $total = 0;// مكان نحسب فيه المجموع
    while( $x <= 100 )
    {
        $total = $total + $x;
        $x++;
    }
    echo " الناتج هو ";
    echo "<br>";
    echo $total;
?>
```

شكل (٣٥) كود PHP لاستخدام جملة التكرار While

(٢)جملة For

يمكن استخدام جملة For في تنفيذ المثال السابق، لطباعة الأعداد من ١ : ١٠٠ كما يلي:

```
<?PHP
    For ($x = 1; $x <= 100; $x ++)
      {
        Echo ($x);
        Echo ("<Br>");
       }
?>
```

حيث:

$x = 1: بداية العداد المتغير $x يساوي قيمة البداية ١.

$x <= 100: يتم تنفيذ جمل التكرار طالما أن $x أقل من أو يساوي ١٠٠، ويتوقف التكرار إذا كانت قيمة المتغير $x أكبر من ١٠٠.

$x ++: زيادة قيمة المتغير $x بمقدار ١. ثم يتم تكرار الكود بين القوسين **{ }**.

(٣) Do … While

ونستطيع أيضًا استخدام جملة **Do … While** لتنفيذ المثال السابق، لطباعة الأعداد من ١ : ١٠٠ كما يلي:

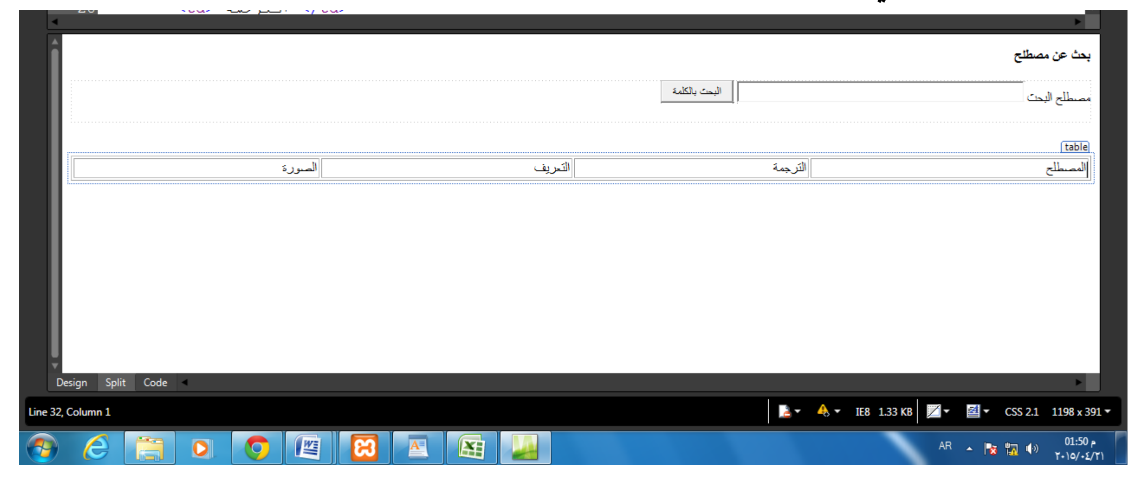
```
<?PHP
    $x = 1;                      // متغير نضع فيه بداية العداد
    Do                           // جملة نفذ ما يلي
```

```
    {                                   //  بداية جمل التكرار
      Echo ($x . "<Br>");               // طباعة المتغير والنزول إلى السطر التالي
      $x ++ ;                           // زيادة قيمة متغير العداد بقيمة ١
    }                                   // نهاية جمل التكرار
  While ($x < = 100);          //  شرط التوقف عن التكرار إذا زاد
                                        متغير العداد عن ١٠٠ يتوقف.
?>
```

خطوات إنشاء صفحة البحث عن مصطلح

(١) تصميم واجهة صفحة البحث عن مصطلح

الغرض من هذه الصفحة البحث عن مصطلح داخل قاعدة البيانات في المصطلحات التي تم إدخالها في جدول قاعدة البيانات "**dbdictionary**" باستخدام جملة **Select**، والشكل التالي يوضح تصميم صفحة البحث في تطبيق إنشاء صفحات الويب:



شكل (٣٦) تصميم واجهة صفحة البحث عن مصطلح

وفي الشاشة السابقة يمكن استنتاج الآتي:

- الصورة **Banner** والارتباطات بباقي صفحات الموقع في بداية صفحة البحث عن مصطلح.
- عنوان صفحة الويب "بحث عن مصطلح".
- تم إدراج **Form** من نافذة **Toolbox**، بحيث يحتوي على عناصر التحكم بصفحة البحث.
- نص "مصطلح البحث" وبجواره مربع النص الذي ندخل فيه المصطلح المراد البحث عنه.
- زر أمر "البحث بالكلمة" والمسئول عن تنفيذ كود **PHP** الخاص بتنفيذ عملية البحث عن المصطلح في قاعدة البيانات، وهو ما يتطلب قبل ذلك الاتصال بقاعدة البيانات.
- إدراج جدول يحتوي على الصف الأول فقط، ويشير إلى عناوين أو رؤوس أعمدة الجدول.

- يمكن استخدام إحدى التطبيقات إنشاء صفحات ويب وذلك من خلال كتابة النصوص وإدراج أدوات التحكم من نافذة **Toolbox** في **Design Mode**، وعرض الكود الذي تم توليده في **Code Mode** بحيث يصبح شكل صفحة الويب على مستعرض الإنترنت للبحث عن مصطلح في قاعدة البيانات كما يلي:



شكل (٣٧) عرض صفحة البحث عن مصطلح على شاشة مستعرض الإنترنت

- نتائج البحث تعرض في الجدول من خلال إضافة صف يُعرض فيه بيانات كل مصطلح.
- يمكن الاطلاع على الكود الخاص بإنشاء الجدول، وذلك بتحديد الجدول، وعرض كود **HTML** الذي تم توليده من خلال الضغط على زر **Code** وهو:

الإعلان عن جدول نسبة عرض الجدول ١٠٠% من صفحة المستعرض وسمك خط الإطار الذي يحيط بالجدول ١.

```
<table style='width: 100%' border="1">
```

<tr>: للإعلان عن صف.

```
<tr>
    <td> المصطلح </td>
```

<td> الإعلان عن العمود الأول حيث يحتوي على كلمة "مصطلح"، وإغلاقه **</td>**.

وهكذا باقي الأعمدة الأربع.

```
    <td> الترجمة </td>
    <td> التعريف </td>
    <td> الصورة </td>
```

إغلاق كود الصف.	</tr>

مع ملاحظة أنه: سيتم إكمال الجدول من خلال كود **PHP** عند عرض نتائج البحث، وحسب عدد الصفوف أو السجلات التي ينطبق عليها شرط البحث.

```
1 <html dir="rtl">
2     <head>
3     <meta content="text/html; charset=utf-8" http-equiv="Content-Type" >
4         <title>بحث عن مصطلح</title>
5     </head>
6 <body>
7
8     <p class="style1"><span lang="ar-eg"><strong>بحث عن مصطلح</strong></span></p>
9     <form method="post" action="search_term.php">
10    <div >
11        مصطلح البحث
12     <input name="txt_search" type="text" style="width: 332px; height: 25px;" >
13     <input name="submit1" type="submit" value="البحث بالكلمة" style="height: 26px" > <br ><br>
14     </form>
15     <br>
16     <table style='width: 100%' border="1">
17     <tr>
18             <td> المصطلح </td>
19             <td> الترجمة </td>
20             <td> التعريف </td>
21             <td> الصورة </td>
22     </tr>
23
24     </table>
25     </div>
26 </body>
27 </html>
```


شكل (٣٨) كود HTML الخاص بصفحة البحث عن مصطلح

(٢) كتابة كود PHP

وفيما يلي نستعرض كود **PHP** رقم ١ الخاص بإضافة الكود في صفحة **Header** وأيضًا كود الاتصال بقاعدة البيانات في صفحة **Connection** في أول جزء من **Body**:

```
<?php
    include("header.php");
    include("connection.php");
?>
```

شكل (٣٩) كود PHP الخاص بتضمين صفحتي Connection & Header

وأيضًا كود **PHP** رقم٢، والخاص بالبحث في قاعدة البيانات عن مصطلح معين حيث يتم إضافته ثم عرضه بالجدول.

```php
<?php
    if(isset($_POST['submit1']))
        {
        $txt_search=$_POST['txt_search'];
        mysql_query("SET NAMES 'utf8'");
        $sql="select * from terms where term like '%$txt_search%' ORDER BY term ASC";
        $query=mysql_query($sql) ;
        $num=mysql_num_rows($query);
        while($row=mysql_fetch_array($query))
            {
                $term=$row['term'];
                $trans=$row['trans'];
                $defe=$row['defe'];
                $picture=$row['picture'];
                echo("        <tr>
                    <td>$term</td>
                    <td>$trans</td>
                    <td>$defe</td>
                    <td><input name='pic' type='image' src='$picture' width='80' height='80'  /></td>
                    </tr>
                ");
            }
        }
?>
```

شكل (٤٠) كود PHP الخاص بالبحث عن مصطلح

<?php

التحقق من الضغط على زر Submit

– اختبار إن كان تم الضغط على زر "**البحث بالكلمة**":

```php
if (isset($_POST['submit1']))
 {
```

– يلتقط المصطلح أو القيمة التي تم إدخالها في مربع النص على شاشة مستعرض الإنترنت، ويضعها في المتغير **$txt_search**.

```php
$txt_search=$_POST['txt_search'];
```

– **استخدام** دالة **mysql_query** يساعد في إظهار البيانات على صفحة المستعرض باللغة العربية بشكل صحيح وعدم ظهورها في صورة علامات استفهام هكذا ؟؟؟؟؟؟؟.

```php
mysql_query("SET NAMES 'utf8'");
```

دالة تنفيذ الاستعلام

– استخدام جملة **Select** في البحث عن جميع حقول جدول البيانات "**term**" باستخدام المعامل "**Like**" وأيضًا يتم وضع علامة "**%**" قبل وبعد المتغير الذي يحتوي على القيمة التي نبحث عنها **%$txt_search%** لعرض البيانات التي تحتوي على الكلمة بالكامل أو جزء منها، وترتيب عرضها تصاعديًا وحفظ الناتج في المتغير **$sql** بينما تشير **ORDER BY** لترتيب الناتج تصاعدي **Ascending** أو تنازلي **Descinding**.

```
$sql="select * from terms where term like '%$txt_search%' ORDER BY
term ASC";
```

– استخدام الدالة **mysql_query** لحفظ ناتج الاستعلام بالمتغير **$sql** في المتغير **$qurey**.

```
$query=mysql_query($sql) ;
```

دالة عد السجلات في الاستعلام

– وأخيرًا يتم عَد السجلات في الاستعلام **$qurey** بالدالة **mysql_num_rows** وحفظها في المتغير **$num**.

```
$num=mysql_num_rows($query);
```

دالة استدعاء سجلات جدول البيانات

– استخدام جملة التكرار **while** لتنفيذ الكود بين القوسين **{ }** حسب عدد السجلات الناتج في الاستعلام بحيث يتم استدعاء أول سجل بيانات من جدول قاعدة البيانات بالدالة **mysql_fetch_array** والانتقال بالمؤشر إلى السجل التالي، ووضعها في المتغيرات الأربع التالية، ويتوقف التكرار عندما لا يتوفر سجل بيانات في المتغير **$row**.

عرض كل سجل في الاستعلام في صف بالجدول على صفحة مستعرض الإنترنت

```
while ($row=mysql_fetch_array($query))
  {
```

– تخصيص الأربع قيم بالسجل الأول **$row** من ناتج الاستعلام للمتغيرات **($term – $trans – $defe – $picture)** في الأربع جمل التالية:

```
$term=$row['term'];
$trans=$row['trans'];
$defe=$row['defe'];
$picture=$row['picture'];
```

فمثًلا: تم تخصيص قيمة الحقل ['term'] بالسجل $row للمتغير $term وهكذا لباقي المتغيرات.

– طباعة ناتج المتغيرات الأربعة في صف جديد داخل الجدول السابق على صفحة مستعرض الإنترنت.

– لكتابة كود **HTML** داخل لغة **PHP** نستخدم الأمر **echo(" ")**.

```
echo("<tr>
            <td>$term</td>
```

```
<td>$trans</td>
<td>$defe</td>
<td>    <input    name='pic'
type='image'    src=    '$picture'
width='80'    height='80'    />
</td>
</tr> ");
```

– إغلاق القوس } التالي يعني العودة إلى جملة **While** لتكرار ما سبق حتى الانتهاء من جميع السجلات في ناتج الاستعلام **$query**.

```
}
```

– إغلاق القوس } التالي يعني إغلاق جملة **IF** في بداية كود **PHP**.

```
}
?>
```

نشاط١: قد يرغب المستخدم في البحث عن مصطلح سبق تخزينه في قاعدة البيانات وعرضه علـى صـفحة الويـب، تحـت اشـراف معلمـك قـم "بإنشـاء صـفحة البحـث عـن مصـطلح "search_term.php من خلال تنفيذ نشاط ١ بكتاب الأنشطة والتدريبات .

(٣) تنفيذ صفحة البحث عن مصطلح

وعند عرض صفحة البحث وإدخال كلمة أو مصطلح مثل كلمة "كمبيوتر"، كما بالشاشة التالية من خلال العنوان **localhost/dictionary/search.php**:



شكل (٤١) استعراض صفحة البحث عن مصطلح على نافذة مستعرض الإنترنت

وبالضغط على زر "البحث بالكلمة" يتم تنفيذ كود **PHP** وتظهر نتيجة البحث في جدول على صفحة المستعرض كما يلي:



شكل (٤٢) تنفيذ البحث عن مصطلح

الموضوع الرابع

صفحة حذف مصطلح Del_term.php

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم واجهة صفحة حذف مصطلح.

٢- ينشيء صفحة حذف مصطلح بكود **HTML**.

٣- يكتب كود **PHP** لتضمين صفحتي **Header & Connection**.

٤- يتناقش مع زملاءه فى كتابة كود **PHP** لحذف مصطلح.

٥- يحذف سجل من قاعدة البيانات.

٦- يتحقق من حذف السجل فى قاعدة البيانات.

يمكن كتابة عنوان الصفحة وإدراج جدول نكتب في صفه الأول عناوين الأعمدة كما يظهر بالشاشة، وبالضغط على زر "**Split**" يمكن عرض الصفحة والكود الذي تم تولده لإنشائها كما يلي:




```
1 <html dir="rtl">
2 <head>
3 <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
4 <title>حذف مصطلح </title>
5 </head>
6 <body>
7
8     <form method="post" action="Del_Term.php">
9     <h1>حذف مصطلح</h1>
10
11        <table style="width: 100%">
12            <tr >
13                <td>المسلسل</td>
14                <td>المصطلح</td>
15                <td>الترجمة</td>
16                <td>التعريف</td>
17                <td>الصورة</td>
18                <td>الخصائص</td>
19            </tr>
20
21        </table>
22
23    </form>
24 </body>
25 </html>
```

شكل (٤٣) كود HTML الخاص بإنشاء صفحة حذف مصطلح

بدراسة الكود السابق يتضح أنه: لا يتضمن أية أوامر جديدة حيث سبق وتناولناها بالشرح، وتشير الأرقام على الشاشة إلى الأماكن التي سوف نقوم فيها بإدراج كود PHP لإجراء المعالجة وتنفيذ عملية حذف مصطلح من جدول قاعدة البيانات.

وفيما يلي نعرض كود **PHP** المطلوب إضافته في الأقسام الأربعة المُشار إليها على الشاشة حيث يمكن إضافتها في الأماكن المحددة بالترتيب.

(٢) كتابة كود PHP

الجزء الأول: حيث يشير الرقم (١) على الشاشة

```
<?php
```

تضمين صفحتي Header والاتصال بقاعدة البيانات

```
include("header.php");
include("connection.php");
mysql_query("SET NAMES 'utf8'");
```

عرض جميع سجلات جدول البيانات terms

استخدام جملة **Select** لعرض جميع سجلات جدول **Terms** وتخصيصها للمتغير **$SQL**.

```
$sql="select * from terms";
$query=mysql_query($sql);
$num=mysql_num_rows($query);
?>
```

- الكود السابق سبق تناوله بالشرح.

الجزء الثاني: حيث يشير الرقم (٢) على الشاشة

```
<?php
```

عرض عدد المصطلحات في جدول البيانات

```
echo ("<h1> عدد المصطلحات $num </h1>");
?>
```

- الكود السابق يطبع النص "عدد المصطلحات" وبجواره ناتج الدالة **mysql_num_rows** التي تم تنفيذها على ناتج الاستعلام **$query** والذي يمثل عدد السجلات التي تحتوي على كلمة معينة.

الجزء الثالث: حيث يشير الرقم (٣) على الشاشة

```
<?php
```

عرض جميع السجلات على صفحة مستعرض الإنترنت

- جملة التكرار **While** تطبق الدالة **mysql_fetch_array** على نتيجة الاستعلام عن كلمة معينة في المتغير **$query**، للحصول على سجل البيانات الذي ينطبق عليه الشرط وتخصيص الناتج في المتغير **$row**.

```
while($row = mysql_fetch_array($query))
```

```
            {
```

– إدراج جدول على شاشة مستعرض الإنترنت يحتوي على الحقول الستة الموضحة على شاشة المستعرض السابقة.

```
            echo ("
              <tr >
                  <td>$id </td>
                  <td>$term </td>
                  <td>$trans </td>
                  <td>$defe </td>
```

– إدراج صورة المصطلح بأبعاد معينة للطول والعرض.

```
                  <td>    <input    name='termp'    type='image'    src='$picture'
                          width='80' height='80'  /> </td>
```

– جعل النص "**حذف**" ارتباط تشعبي، بحيث يستدعي صفحة حذف المصطلح عند الضغط على كلمة "**حذف**" للمعامل **id** الذي يحتوي على رقم السجل **$id** المراد حذفه.

```
                  <td> <a href='delete_term.php?id=$id'>حذف </a> </td>
              </tr>
```

– **إغلاق كود Echo.**

```
        ");
```

– إغلاق الحلقة التكرارية **While**.

```
        }
?>      //        PHPإغلاق كود
```

الجزء الرابع: حيث يشير الرقم (٤) على الشاشة

```
<?php
```

تنفيذ حذف السجل

– عند الضغط على الارتباط التشعبي "**حذف**" يتم استدعاء صفحة الحذف و **id**، كما يتم التقاط **id** باستخدام **'$_GET[id]'** وبذلك يتم حذف السجل.

```
      if (isset($_GET['id']))
         {
```

– تخصيص جملة **SQL** الخاصة بحذف السجل الذي رقمة **ID** في متغير حرفي باسم **$sqldel**.

```
$sqldel="delete from terms where id='$_GET[id]'";
```

– تنفيذ جملة **SQL** السابقة، حيث يكون ناتج التنفيذ **True** أو **False** في المتغير **$query**.

```
$query=mysql_query($sqldel);
```

– إعادة توجيه المستعرض **Redirect** لصفحة الحذف.

```
header("Location: delete_term.php");
exit;
    }
?>
```

يمكنك استخدام صفحة الويب كواجهة لحذف سجل بيانات مصطلح معين من جدول قاعدة البيانات من خلال كود PHP ، وذلك بتنفيذ النشاط ١ بكتاب الأنشطة والتدريبات

(٣) تنفيذ صفحة حذف مصطلح

عند تحميل صفحة حذف مصطلح معين، يتم عرض المصطلحات وجميع بياناتها، كما بالشاشة التالية من خلال العنوان **localhost/dictionary/del_term.php**:

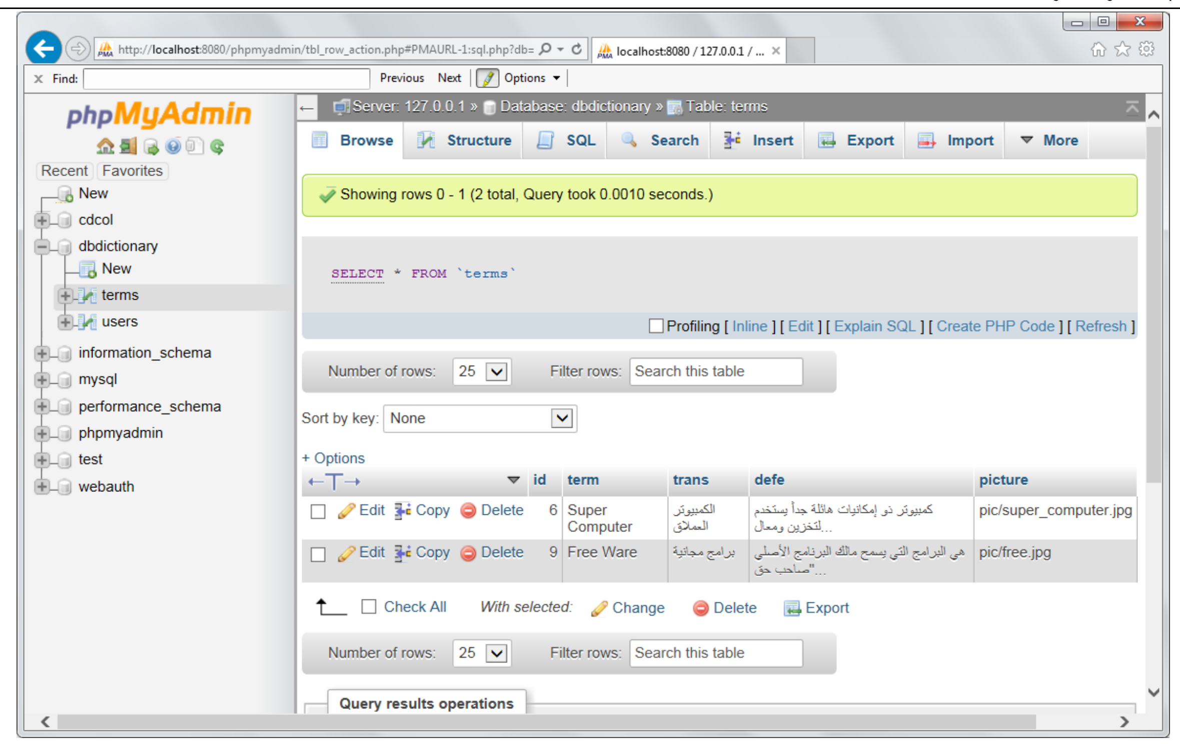


شكل (٤٤) تنفيذ عملية حذف مصطلح

(٤) التأكد من حذف السجل في قاعدة البيانات

يمكن فتح قاعدة البيانات في تبويب جديد لمستعرض الإنترنت، ومشاهدة السجل المراد حذفه قبل الضغط على رابط "حذف" المصطلح حيث تظهر قاعدة البيانات كما يلي من خلال الرابط التالي:

"Localhost/phpmyadmin"



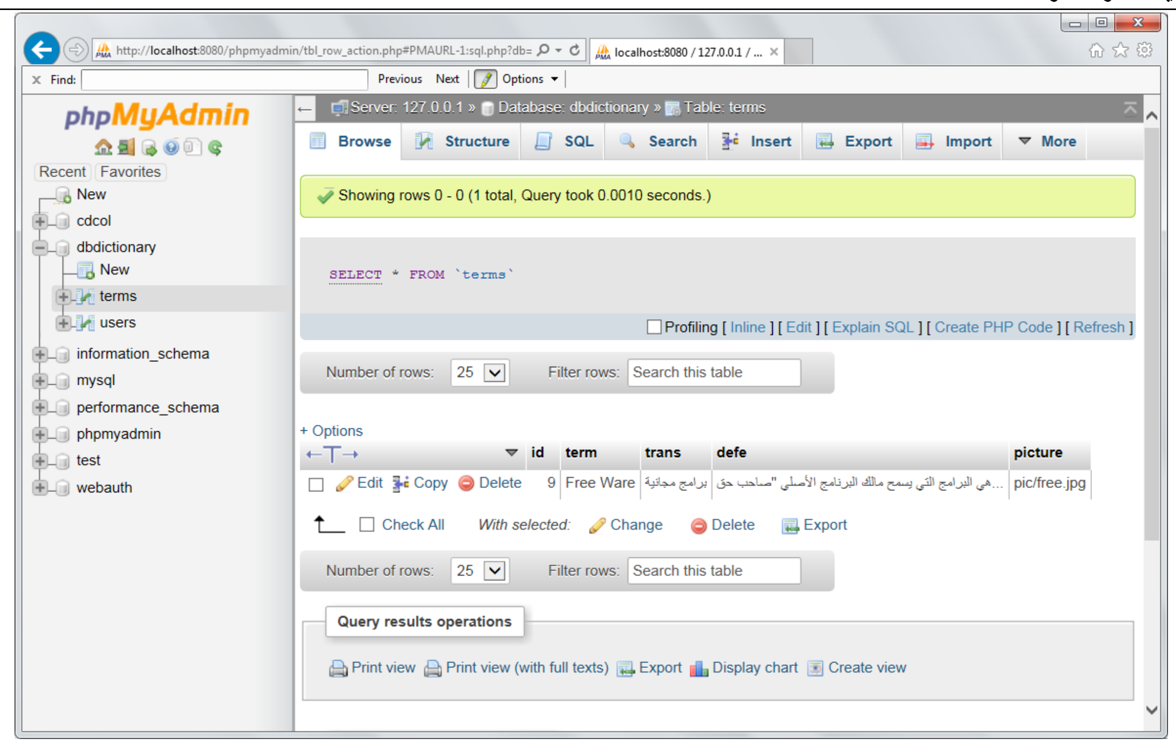
شكل (٤٥) التأكد من وجود المصطلح في جدول قاعدة البيانات قبل حذفه

وبالضغط على الرابط "حذف" في صفحة حذف مصطلح يتم حذف السجل المجاور له وتظهر شاشة مستعرض الإنترنت كما يلي:



شكل (٤٦) التأكد من حذف المصطلح على نافذة مستعرض الإنترنت

وللتأكد من حذف السجل أو بيانات مصطلح معين من قاعدة البيانات، نفتح قاعدة البيانات في تبويب جديد لمستعرض الإنترنت، ونتأكد من حذف السجل من جدول "**terms**":



شكل (٤٧) التأكد من حذف المصطلح من جدول قاعدة البيانات

الموضوع الخامس

صفحة "تعديل بيانات مصطلح Edit_term.php"

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم واجهة صفحة تعديل مصطلح.

٢- ينفذ كود **HTML** لإنشاء صفحة تعديل مصطلح.

٣- يكتب كود بلغة **PHP** لصفحتى **Header & Connection**.

٤- يكتب كود بلغة **PHP** لإجراء تعديل بيانات مصطلح.

٥- يعدل بيانات فى جدول قاعدة البيانات

٦- يتحقق من تعديل بينات مصطلح فى قاعدة البيانات.

٧- يحدد العناصر التى يمكن عرضها فى صفحة المساعدة.

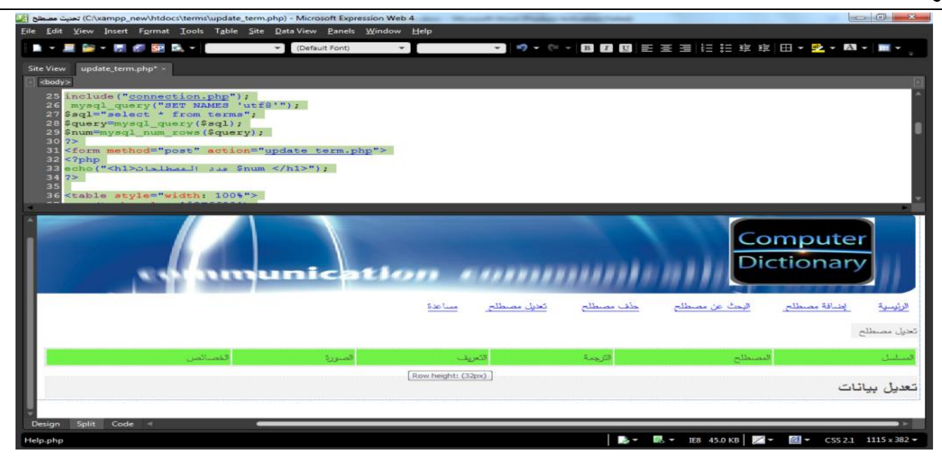
٨- ينشىء صفحة المساعدة **Help**.

الغرض من هذه الصفحة تعديل بيانات مصطلح وتخزين البيانات المُعدلة في قاعدة البيانات، ولإجراء تعديل في بيانات مصطلح نحتاج للقيام بما يلي:

- عرض المصطلحات.
- اختيار المصطلح المراد تعديله.
- عرض بيانات المصطلح المراد تعديله في عناصر التحكم على صفحة المستعرض.
- إجراء التعديل داخل قاعدة البيانات.
- عرض البيانات بعد التعديل.

(١) تصميم واجهة صفحة التعديل

وذلك من خلال إحدى تطبيقات إنشاء صفحات الويب، حيث يمكن كتابة عنوان الصفحة وإدراج جدول نكتب في صفه الأول عناوين الأعمدة كما يظهر في الشاشة التالية:



شكل (٤٨) تصميم واجهة صفحة تعديل مصطلح

إجراء المعالجة وتعديل بيانات مصطلح وحفظ التعديلات في جدول قاعدة البيانات يمكن التعرف على أماكن إضافة كود **PHP** اللازم لإجراء المعالجة، والموضحة بالشاشة التالية:

```
1  <html  dir="rtl">
2      <head>
3      <meta content="text/html; charset=utf-8" http-equiv="Content-Type" >
4      <title>تحديث مصطلح </title>
5      </head>
6  <body>
7  <!--    php code number    1   -->
8    <h1><p  > تعديل مصطلح </p>     </h1>
9  <!--    php code number    2  -->
10     <form method="post" action="edit_term.php">
11 <!--    php code number    3   -->
12         <table style="width: 100%">
13             <tr bgcolor='#3Fff00'>
14                 <td>المسلسل</td>
15                 <td>المصطلح</td>
16                 <td>الترجمة</td>
17                 <td>التعريف</td>
18                 <td>الصورة</td>
19                 <td>الخصائص</td>
20             </tr>
21 <!--    php code number    4   -->
22          </table>
23 <!--    php code number    5   -->
24      </form>
25     <form method='post' action='edit_term.php?id=$_GET[id]&amp;action=$_GET[action]' enctype="multipart/form-data">
26     <hr  style='color:orange;maroon;width:1267px'/>
27     <h2>تعديل بيانات </h2>
28     <hr  style='color:orange;maroon;width:1267px'>
29          <div >
30  <!--    php code number    6    -->
31
32 <!--    php code number    7   -->
33
34          </div>
35      </form>
36 </body>
37 </html>
```

كود PHP رقم ١

كود PHP رقم ٢

كود PHP رقم ٣

كود PHP رقم ٤

كود PHP رقم ٥

كود PHP رقم ٦

كود PHP رقم ٧

شكل (٤٩) كود HTML الخاص بإنشاء صفحة تعديل مصطلح

الكود السابق لا يتضمن أية أوامر جديدة حيث سبق وتناولناها بالشرح، وتشير الأرقام على الشاشة إلى الأماكن التي سوف نقوم بإدراج كود **PHP** فيها لإجراء المعالجة وتنفيذ عملية تعديل بيانات مصطلح في جدول قاعدة البيانات، وفيما يلي نعرض كود **PHP** المطلوب إضافته في الأقسام الستة المُشار إليها على الشاشة حيث يمكن إضافتها في الأماكن المحددة بالترتيب:

وقبل البدء في دراسة كود تعديل بيانات مصطلح يمكن عرض الفكرة العامة للتعديل من خلال الجدول التالي:

الرقم	الغرض من الكود
١	تضمين صفحة **Header**.
٢	تحقيق الاتصال بقاعدة البيانات، وضبط عرض اللغة العربية، وتخزين البيانات من جدول قاعدة البيانات، ثم نقل ناتج الاستعلام **mysql_query($sql)** في متغير **$query**، وأخيرًا استخدام الدالة **mysql_num_rows** لحصر عدد السجلات المراد تعديلها وحفظها في المتغير **$num**.
٣	عرض عدد السجلات المتاح تعديلها.
٤	البحث عن أول سجل وحفظ بياناته في متغيرات وعرضها في صف جديد من صفوف جدول عرض البيانات على شاشة مستعرض الإنترنت، وجعل كلمة "تعديل" ارتباط تشعبي بصفحة تعديل مصطلح.
٥	من خلال كود المصطلح **Id** يتم البحث عن السجل بالكامل وتخزين بيانات المصطلح في المتغيرات.
٦	إضافة عناصر تحكم أسفل الجدول على شاشة مستعرض الإنترنت، وعرض البيانات المراد تعديلها فيه وبعد إجراء المستخدم لكافة التعديلات،
٧	بالضغط على زر "إضافة" في جدول قاعدة البيانات يتم استبدال بيانات المصطلح بالبيانات المُعدلة على شاشة مستعرض الإنترنت وعرضها بعد التعديل على شاشة مستعرض الإنترنت.

(٢) كتابة كود الصفحة

الجزء الأول: حيث يشير الرقم (١) إلى مكان إضافة الكود التالي بعد <Body>

```
<?php
        Include ("header.php");
?>
```

– الكود السابق سبق تناوله بالشرح.

الجزء الثاني: حيث يشير الرقم (٢) إلى مكان إضافة الكود التالي قبل<Form>

```
<?php
     include("connection.php");
     mysql_query("SET NAMES 'utf8'");

                                                  عرض جميع سجلات جدول Term
     $sql="select * from terms";
```

```
     $query=mysql_query($sql);
     $num=mysql_num_rows($query);
?>
```

- الكود السابق سبق تناوله بالشرح.

الجزء الثالث: حيث يشير الرقم (٣) إلى مكان إضافة الكود التالي بعد نص "عدد المصطلحات".

```
<?php
عرض عدد المصطلحات في جدول البيانات
        echo("<h1> عدد المصطلحات $num </h1>");
?>
```

- **الكود السابق يطبع نص عدد المصطلحات وبجوارها قيمة المتغير $num** الذي يحتوي على عدد المصطلحات.

الجزء الرابع: حيث يشير الرقم (٤) إلى مكان إضافة الكود التالي قبل إغلاق الجدول </table>.

```
<?php
اكمال عملية عرض جميع السجلات في عناصر التحكم على صفحة المستعرض
```

- استخدام جملة التكرار **while** لتنفيذ الكود بين القوسين **{ }** حسب عدد السجلات الناتج في الاستعلام بحيث يتم استدعاء أول سجل بيانات من جدول قاعدة البيانات بالدالة **mysql_fetch_array** والانتقال بالمؤشر إلى السجل التالي، ووضعها في المتغيرات الخمسة التالية ويتوقف التكرار عندما لا يتوفر سجل بيانات في المتغير **$row**.

```
        While ($row = mysql_fetch_array($query))
         {
           $id=$row['id'];
           $term=$row['term'];
           $trans=$row['trans'];
           $defe=$row['defe'];
           $picture=$row['picture'];
```

- **إدراج صف جديد للجدول على شاشة المستعرض يحتوي على قيمة المتغيرات الخمسة الخاصة بالمصطلح.**

```
           echo ("
```

```
        <tr >
            <td>$id</td>
            <td>$term</td>
            <td>$trans</td>
            <td>$defe</td>
            <td> <input name='termp' type='image'src='$picture' width='80'
                height='80'/>
            </td>
```

- ربط النص "تعديل" بصفحة تعديل مصطلح.

```
            <td> <a href='update_term.php?id=$id'>تعديل</a> </td>
        </tr>
            ");
    }
?>
```

الجزء الخامس: حيث يشير الرقم (٥) إلى مكان إضافة الكود التالي بعد إغلاق الجدول </table>.

```
<?php
```

تخصيص قيم حقول السجل المراد تعديله عند النقر على رابط "تعديل"

- عند الضغط على زر تعديل بجوار سجل معين يتم الحصول على رقمه Id باستخدام $_GET['id'] وذلك من خلال جملة Select حيث يحفظ الناتج في المتغير $sql.

```
    If (isset($_GET['id']))
     {
      $sql="select * from terms where id='$_GET[id]'";
```

- ومن خلال الدالة mysql_query يتم البحث عن السجل في المتغير $sql وحفظه في المتغير $query.

```
      $query=mysql_query($sql);
```

- ثم يتم تخصيص جميع الحقول في الصف أو السجل المراد تعديله للمتغيرات التالية على الترتيب ($id – $term – $trans – $defe – $picture).

```
      $row=mysql_fetch_array($query);
      $id=$row['id'];
      $term=$row['term'];
      $trans=$row['trans'];
      $defe=$row['defe'];
```

```
$picture=$row['picture'];
?>
```

الجزء السادس: حيث يشير الرقم (٦) إلى مكان إضافة الكود التالي قبل إغلاق الـ </Form>.

```
<?php
```

إنشاء عناصر التحكم على صفحة المستعرض وضبط بعض خصائصها ووضع القيم في المتغيرات الخاصة بالحقول لكل عنصر تحكم مقابل له

```
Echo ("
        <br />
```

– استخدام input في إدراج عناصر تحكم على شاشة مستعرض الإنترنت تسمح للمستخدم بإدخال البيانات الجديدة أو المعدلة للمصطلح، ويحدد في هذا الكود اسم عنصر التحكم ونوعه وعرضه والقيمة التي يحتويها من المتغير المناظر له في الجزء الخامس.

```
<input name='iddata' type='hidden' style= 'width: 482px' value='$id'/>
```

نلاحظ أنه يمكن استبدال قيمة عنصر التحكم value='$id' في سطر الكود السابق بالكود <?php echo "$id" ?> المناظر له في لغة PHP.

```
المصطلح   
<input name='txt_term' type='text' style= 'width: 482px'
 value='$term' /> <br/> <br/>
ترجمة   
<input name='trans' type='text' style= 'width: 482px'
 value='$trans' /> <br/><br/>
تعريف   
<textarea name='TextArea1' style='width: 480px; height: 30px'>
 $defe</textarea> <br/>
<br/>  الصورة   
<input name='termp' type='image' src='$picture' width='80'
 height='80' />
```

```
<input name='pic' type='hidden' style='width: 482px'
  value='$picture' />
<input name='filedata' type='file' maxlength='43'>
<br/> <br/> <br/> <br/>
```

– إدراج زر "إضافة" نوعه Submit2، حيث يقوم بتنفيذ الكود في السطور التالية عند الضغط عليه Click.

```
<input name='Submit2' style='width: 76px' type='submit'
 value='إضافة' /> <br/> <br/>
  ");
```

الجزء السابع:

إجراء تعديل بيانات مصطلح بكتابة كود php رقم ٧

– الكود التالي يختبر إن كان المستخدم قد قام بالضغط Click على زر "إضافة" فيتم التحقق أوّلا من وجود مجلد باسم "Pic" فإذا لم يكن موجودًا يتم إنشائه، ثم يتم التأكد من وجود قيم في عناصر التحكم وأنها غير خالية، وأيضًا تخزين البيانات الجديدة بعد التعديل في متغيرات.

```
If (isset($_POST['Submit2']))
  {
   If (!is_dir('pic'))
     {
      mkdir(pic);
     }
  $fileName = $_FILES['filedata']['name'];
  $tmpName  = $_FILES['filedata']['tmp_name'];
  If (!empty($fileName))
    {
      move_uploaded_file($tmpName,'pic/'.$fileName);
    }
$iddata=$_POST['iddata'];
$terma=$_POST['txt_term'];
$transa=$_POST['trans'];
```

```
$defea=$_POST['TextArea1'];
$pic=$_POST['pic'];
```

إذا لم يتم تغيير صورة المصطلح تستخدم الصورة السابقة.

```
If (!empty($fileName))
  {
   $picturea="pic/" . $fileName . " ";
   }
else{
        $picturea= $pic ;
        }
 mysql_query("SET NAMES 'utf8'");
```

– الكود التالي لتحديث البيانات في جدول قاعدة البيانات بالقيم الجديدة التي تم حفظها في المتغيرات الأخيرة.

```
$sql="update terms set term='$terma',trans='$transa'   ,defee =
'$defea'    ,picture='$picturea'      where id='$iddata' ";
$query=mysql_query($sql);
```

عرض الصفحة بعد التعديل

```
If ($query)
  {
    header("Location: update_term.php");
    exit;
  }
      }
   }
?>
```

تحت اشراف معلمك استخدم الشاشتين المعروضتين بنشاط ١ في إنشاء صفحة تعديل بيانات مصطلح edit_term.php - كتاب الأنشطة والتدريبات ص

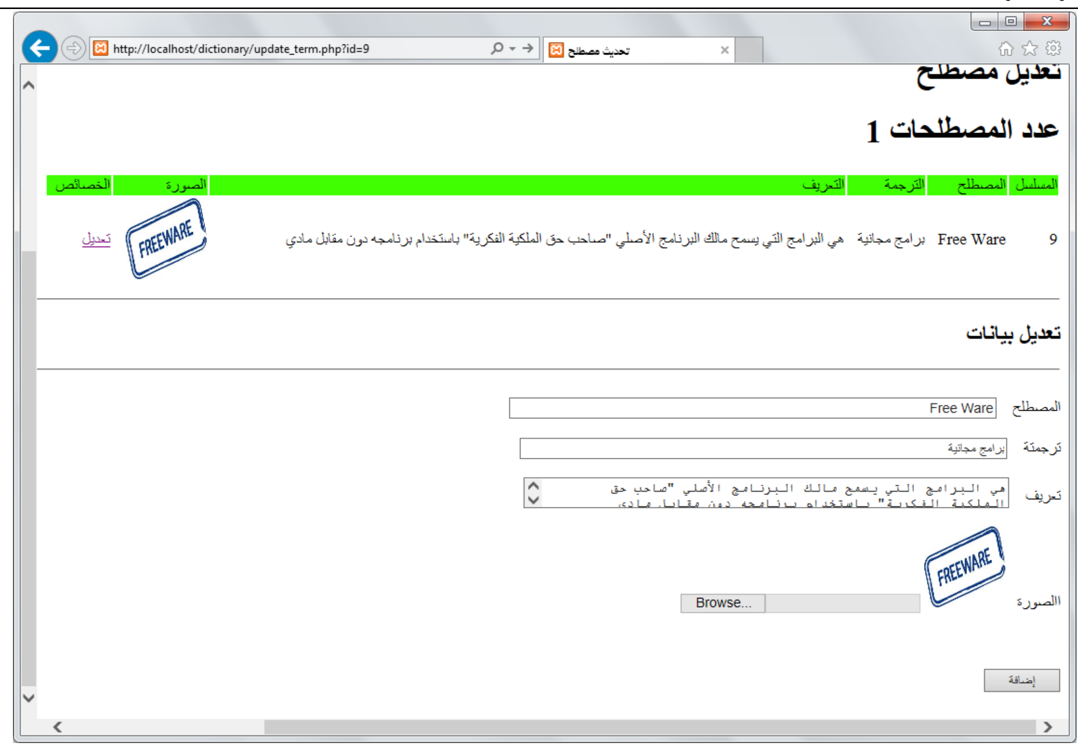
(٣) تنفيذ صفحة تعديل بيانات مصطلح

عند تحميل صفحة تعديل بيانات مصطلح معين، يتم عرض المصطلح المراد تعديله وجميع بياناته، كما بالشاشة التالية وذلك على الرابط **localhost/dictionary/edit_term.php**:



شكل (٥٠) عرض صفحة تعديل مصطلح على نافذة مستعرض الإنترنت

وبالضغط على رابط تعديل يتم تنفيذ كود التعديل وتظهر صفحة مستعرض الإنترنت وبها بيانات المصطلح أسفل الشاشة حيث يقوم المستخدم بتعديل بيانات المصطلح ويضغط زر "إضافة" كما يظهر بالشاشة التالية:



شكل (٥١) تنفيذ تعديل مصطلح على نافذة مستعرض الإنترنت

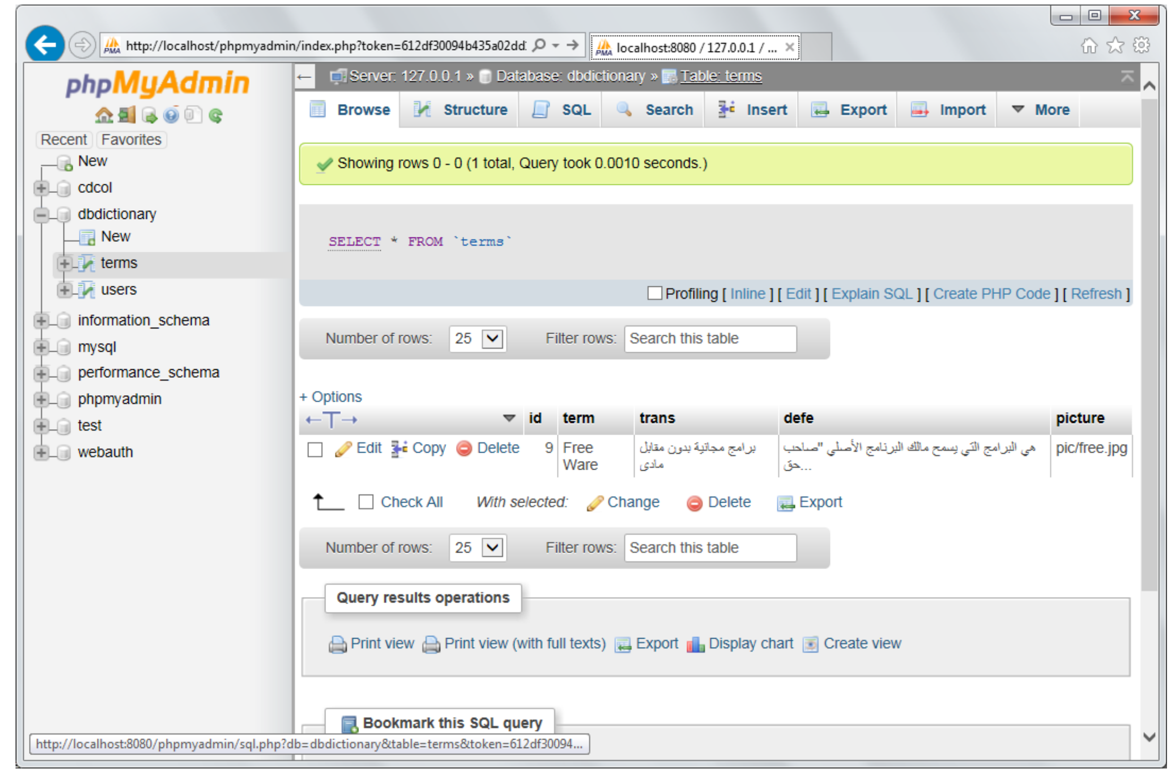
ومع تنفيذ التعديل تظهر بيانات المصطلح بعد التعديل في نفس صفحة مستعرض الإنترنت كما يلي:



شكل (٥٢) عرض صفحة تعديل مصطلح على نافذة مستعرض الإنترنت بعد تنفيذ التعديل

(٤) التأكد من تعديل بيانات المصطلح في قاعدة البيانات

يمكن فتح قاعدة البيانات في تبويب جديد لمستعرض الإنترنت بكتابة الرابط التالي "**localhost/phpmyadmin**" في شريط العنوان، ومشاهدة السجل بعد التعديل حيث تظهر قاعدة البيانات كما يلي:



شكل (٥٣) التأكد من تعديل بيانات المصطلح في جدول قاعدة البيانات

إنشاء صفحة المساعدة Help.php

عادة ما تتضمن المواقع صفحة مساعدة حيث يمكن إنشائها من خلال النشاط التالي.

كراسة الأنشطة: نشاط "إنشاء صفحة المساعدة help.php"

الفصل الدراسي الثاني

الوحدة الرابعة

تأمين موقع "قاموس مصطلحات الكمبيوتر المصور"

فى نهاية الوحدة يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم وسيلة إلكترونية لنشر الوعى بالاستخدام الآمن للإنترنت

٢- يناقش المفاهيم والقضايا المتعلقة بالأمن الفكرى وأمن المعلومات عند استخدام وسائل تكنولوجيا المعلومات والاتصالات.

٣- يمارس أخلاقيات وسلوكيات أحترام حقوق الملكية الفكرية عند التعامل مع المعلومات والأجهزة والشبكات وخدمات وتطبيقات الويب.

٤- يراعى القيم الأخلاقية ويحترم القانون والملكية الفكرية عند استخدام المعلومات والأجهزة والشبكات

٥- يتبع أساليب حماية البرامج التطبيقية من المخاطر التي يمكن ان يتعرض لها

٦- يتبع أساليب حماية المواقع وقواعد البيانات من الأنتهاكات التي يمكن ان يتعرض لها

٧- يوثق المعلومات التي حصل عليها من مصادر المعلومات الإلكترونية

مقدمة

عزيزي الطالب والطالبة تعلمت من خلال الفصول السابقة إنشاء موقع ويب باستخدام لغة **PHP** وقد لاحظت أنه يمكن لأي زائر للموقع أن يقوم بإدخال بيانات للموقع (مصطلحات) أو التعديل أو الحذف، وقد سبق وشرحنا الموقع بهذا الشكل لإكسابك المعلومات والمهارات الأساسية لتصميم مواقع الويب بلغة php .

إذن لابد لك من تامين هذا الموقع ان تكون على دراية باساليب الحماية من اختراق مواقع الويب عمومًا وهذا ما سوف تتعلمه خلال هذه الوحدة.

لذلك سوف نقوم خلال هذه الوحدة بانشاء صفحة تسجيل مستخدم جديد وصفحة للدخول وصفحة للخروج وعمل الاجراءات اللازمة لحماية قاعدة البيانات dbdictionary وكل هذا سيتم بشكل مبسط دون الدخول فى اجراءات التامين الاحترافية المعقدة.

وسوف نبدأ بشرح مدخل لتامين مواقع الويب

الموضوع الأول
مدخل فى تامين مواقع الويب

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يتعرف مفهوم الأختراق.

٢- يشرح طرق حماية مواقع الويب.

٣- يبحث عبر مصادر المعلومات الالكترونية عن طرق وأساليب اختراق مواقع الإنترنت.

يعتبر تأمين مواقع الويب **Secure Websites** ضرورة تفرض نفسها وذلك للحد من اختراقها، والذى يمكن أن ينتج عنه الآتي:

بعض النتائج الضارة لاختراق مواقع الويب:

- فقد البيانات والتي قد تكون هامة وتؤدى إلى خسائر مالية لبعض المؤسسات.
- سرقة بيانات هامة من الموقع .
- الحصول على بيانات مؤسسية أو شخصية وما لهذا من أضرار .
- عرض محتوى آخر غير ملائم قد يحتوى على توجهات سياسية أو دينية أو أخلاقية غير مرغوبة.
- تشويه صورة المؤسسة أو الشخص صاحب الموقع بشكل عام، وبالتالي يؤدي إلى فقدان ثقة المستخدمين والزائرين فيه.

مفهوم الاختراق Penetration

اختراق الموقع **Website Penetration** ويعبر عنه عادةً بـ **Website Hacking** حيث يمكن للمخترق **Hacker** من الحصول على صلاحية التحكم في إدارة الموقع أو التعامل مع بيانات الموقع بأي صورة (عرض – حذف – تعديل ... الخ) من خلال استغلال ثغرة أمنية أو برمجة ضعيفة.

لذلك يجب أن نكون على دراية جيدة بطرق حماية مواقع الويب والتي تنقسم إلى قسمين هما:

١- حماية على مستوى الخادم (Server) (الخادم المستضيف للموقع Website Hosting):

وتكون الحماية هنا مسئولية الخادم أو الجهة المستضيفة للموقع والتي يجب أن تقوم بإعداد خيارات الأمان بشكل صحيح ، كما تتحمل كثير من التحديثات وخصوصا نظم تشغيل الخادم.

٢- حماية على مستوى مطوري الموقع

وتكون الحماية هنا مسئولية مطوري الموقع والمسئولين عن إدارته مثل:

- التحقق من المدخلات قبل تخزينها في قاعدة البيانات.
- تشفير كلمات المرور.
- إدارة مجلدات الموقع الهامة بكلمات سر قوية.
- تحديد صلاحيات للمستخدمين بشكل صحيح.

بعض احتياطات الأمان للحفاظ على تأمين الموقع

١. تحديث البرامج Keep software up to date

يجب التأكد من الحصول على التحديثات الخاصة بالبرامج المستخدمة في إدارة و تصميم الموقع سواء كانت برامج نظم تشغيل الخادم أو أي برامج تعمل على الموقع.

٢. التعامل مع رسائل الخطأ Error messages

من الضرورى التعرف على كافة الأخطاء المحتملة عند نشر الموقع،مثل عدم تحقق الاتصال بقاعدة البيانات أو عدم حفظ المصطلح بالجدول بشكل صحيح لذلك يجب الحرص على إخفاء الأخطاء، حيث أن هذه الأخطاء تجعل الموقع ضعيف وأكثر عرضه للاختراق **ولا يجب ظهورها للمستخدم وانما يتم استبدال الخطأ برسالة يتم عرضها للمستخدم برمجياً**

إذن يتم توقع الخطأ والتعامل معه برمجيًا –مثل جملة **Try ... Catch**– من خلال رسائل خطأ معدة بعنايه ولا توحى للمستخدم بأي معلومات قد تستخدم في الاختراق مثلا عند وجود خطأ في كلمة السر يمكن إعطاء رسالة "اسم المستخدم أو كلمة السر غير صحيحة".

وسوف تتعلم خلال هذه الوحدة مهارة عرض رسالة خطأ برمجيًا فى مواقف مختلفة

٠٣التحقق من صحة البيانات المدخلة من المستخدم (زائر الموقع) Input Data Validation

عدم التحقق من البيانات المدخلة يعطى الفرصة لذوى النوايا السيئة بإختراق الموقع بإدخال مدخلات معينة تتسبب في الاختراق، ولذلك فإن التحقق من صحة البيانات المدخلة من المستخدم من أهم قواعد الحماية من الاختراق **مثًلا** التحقق من أن الحقل يحتوى قيم معينة لا تزيد عن عدد محدد من الأحرف أو التحقق من أن الحقل غير فارغ، وذلك باستخدام جملة **IF**.

يمكن التحقق عن طريق أكواد جافا سكريبت حيث يمكن للمخترق تعطيل برمجة الجافا في المستعرض ولذلك يجب التحقق مرة أخرى عند الخادم باكواد php.

والسبب في ذلك يرجع إلى أن كود **PHP** يتم تنفيذه على الخادم **Server** وليس على جهاز العميل **Clint**.

٠٤ كلمات المرور Passwords

كلمات المرور يجب أن تكون معقدة نوعًا ما حتى يصعب على المخترق اكتشافها وخاصةً كلمة مرور الخادم server وكلمة مرور **admin** الخاصة بالموقع وكلمات مرور قاعدة البيانات.

بالنسبة لكلمات المرور الخاصة بمستخدمي الموقع يمكن الزام المستخدم على إدخال كلمات مرور ذات متطلبات معينة مثل عدد الأحرف لا يقل عن ثمانية – ضرورة وجود حروف كبيرة مع أرقام مع علامات خاصة. ويجب حفظ كلمات المرور **دائمًا أبدًا** مشفرة **Encrypted** باستخدام إحدى أساليب التشفير المتاحة في لغة PHP مثل: دالة SHA (**Salt Password**) أو دالة **MD5**.

٠٥تجنب ادراج/حقن جمل SQL (SQL Injection)

هي محاولة المخترق ادراج أو حقن معامل خاص **Parameter** داخل جملة **SQL** من خلال نموذج إدخال البيانات بالموقع ليتم تنفيذها على قاعدة البيانات بدون علم مصممي ومسئولي الموقع.

أى يتم تغيير جملة Sql لتعطى نتائج اخرى يستغلها النخترق استغلال سىء

مثال: يقوم المخترق بإدخال معامل خاص في جملة **SQL** بدًلا من إدخال اسم المستخدم بهدف أحداث تعديلات غير مرغوبة بجداول قاعدة البيانات.

٠٦تجنب كتابة كود عبر الموقع XSS (Cross Site Scripting)

يقصد بها إدراج كود في صفحات الموقع والخطورة تكمن في قبول هذا الكود لعدم وجود برمجة تحقق من المدخلات.

مثال: بفرض وجود نموذج يسمح للمستخدم بإدخال تعليق **Comment** وعرض التعليقات بعد ذلك بشكل ناجح، يمكن للمخترق إدخال كود جافا سكريبت **Java Script** مثًلا وعند إرسال التعليق إلى الخادم **server** يخزن هذا الكود في قاعدة البيانات وعند عرضه في صفحة **html** يتم تنفيذ هذا الكود والذي ربما يعيد توجيه المستخدم إلى صفحة أخرى ذات محتوى سيء أو صفحة اصطياد **phishing** (أي تحتوى على نموذج إدخال وهمي

للحصول على بيانات هامة من زائر الموقع مثل كلمة سر أو رقم فيزا كارت)، ويمكن تجنب ذلك باستخدام أسلوب البرمجة المناسب (مثل عدم السماح بأي كود **Script** في حقل التعليقات).

٧.رفع الملفات File Uploads

السماح برفع ملفات إلى موقعك يحتوى على مخاطرة كبيرة يجب تفاديها باتباع الاحتياطات البرمجية اللازمة حيث قد يحتوي الملف على كود **script** يتم تنفيذه بمجرد فتح الملف على الخادم وبالتالي يصبح موقعك ضحية للمخترق، ويتم علاج هذا الاحتمال بعمل اختبار للملف الرفوع فإذا كان ملف صورة مثلًا توفر لغة **php** العديد من أساليب البرمجة للتأكد من هوية الملف.

٨. طبقة الاتصال الآمن (Secure Sockets Layer).SSL

هو بروتوكول لتدعيم التعامل الآمن بين خادم الويب **Web Server** ومستعرض الإنترنت **Web Browser** عن طريق توسيط طرف ثالث يسمى **Certificate Authority (CA)**, والتي يمكن ترجمتها جهة التصديق والتي بدورها توفر صفحات آمنة تستخدم بروتوكول **HTTPS** بدًلا من **HTTP** وخصوصًا للمواقع التي بها تعاملات مالية أو نماذج بيانات هامة.

٩.استخدام تطبيقات تأمين مواقع الويب Website Security Tools

بعد الانتهاء من تصميم الموقع يجب اختبار أمان الموقع، والطريقة الفعالة لذلك استخدام تطبيقات أو أدوات أمان الموقع وتسمى أحيانًا اختبار الاختراق **Penetration Testing** أو اختبار القلم **Pen Testing**ويوجد العديد من هذه التطبيقات منها ما هو مجانًا أو مفتوح المصدر، وهذه التطبيقات تقوم باختبار أمان الموقع ضد الاختراق باستخدام أكواد وأساليب مشابهة لما يقوم به المخترقون.

أمثلة من هذه التطبيقات

١- **OpenVAS** ويعتبر من أكثر التطبيقات مفتوحة المصدر استخدامًا لاختبار أمان الموقع.

٢- **Netsparker** وهو جيد لاختبار **SQL injection and XSS**.

نشاط ١:ابحث عبر الإنترنت عن طرق وأساليب اختراق مواقع الإنترنت وأفضل طرق تجنبها.
كتاب الأنشطة والتدريبات ص

وسوف نبدأ الآن بتأمين موقع قاموس وهو مايتم من خلال:

١- تسجيل مستخدمين للموقع وذلك بانشاء (صفحة تسجيل مستخدم جديد **reg.php**) وحفظ بياناتهم في قاعدة البيانات.

٢- ضرورة تسجيل الدخول **LOGIN** للسماح بعمليات الإدخال أو التعديل أو الحذف وذلك بإنشاء (صفحة تسجيل دخول **signin.php**).

٣- تشفير **Encrypt** كلمة المرور الخاصة بأي مستخدم.

٤- عمليات إضافة مصطلح أو تعديل بياناته أو حذف مصطلح يقوم بها فقط زائري الموقع المسجلين المسموح لهم بذلك.

مفهوم Session: تعني "جلسة" وهو طريقة لتخزين بيانات في متغير داخل ذاكرة الخادم Server لمستخدم واحد فقط ويكون هذا المتغير متاحًا للاستخدام خلال كافة صفحات الموقع، وذلك طيلة الوقت الذي يظل فيه المستخدم يتجول في صفحات الموقع.

فمثلًا: عند تصميم صفحة دخول مستخدم يتم التحقق من كون المستخدم قد قام بتسجيل دخول من عدمه حتى يتم استخدام متغيرات الجلسة **Session variables** خلال صفحات الموقع، وذلك من خلال إضافة الكود التالي في بداية أي صفحة بالموقع نريد استخدام متغيرات الجلسة بها.

```
<?php
session_start();
?>
```

شكل (٥٤) كود PHP الخاص بدء الـ Session

الموضوع الثاني
إنشاء صفحة تسجيل مستخدم جديد

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- ينشيء صفحة تسجيل مستخدم جديد.

٢- يصمم صفحة تسجيل مستخدم جديد.

٣- يفرق بين **GET & POST**.

٤- يشرح احتمالات الخطأ فى إدخال اسم المستخدم وكلمة المرور.

٥- يشارك زملاءة فى كتابة كود **PHP** لإنشاء صفحة تسجيل مستخدم جديد.

٦- يبحث عبر مصادر المعلومات الالكترونية عن طرق تأمين كلمة المرور.

اجراءات انشاء صقحة تسجيل مستخدم جديد

أولاً: إنشاء جدول جديد باسم Users في قاعدة البيانات dbdictionary يحتوى على الحقول التالية:

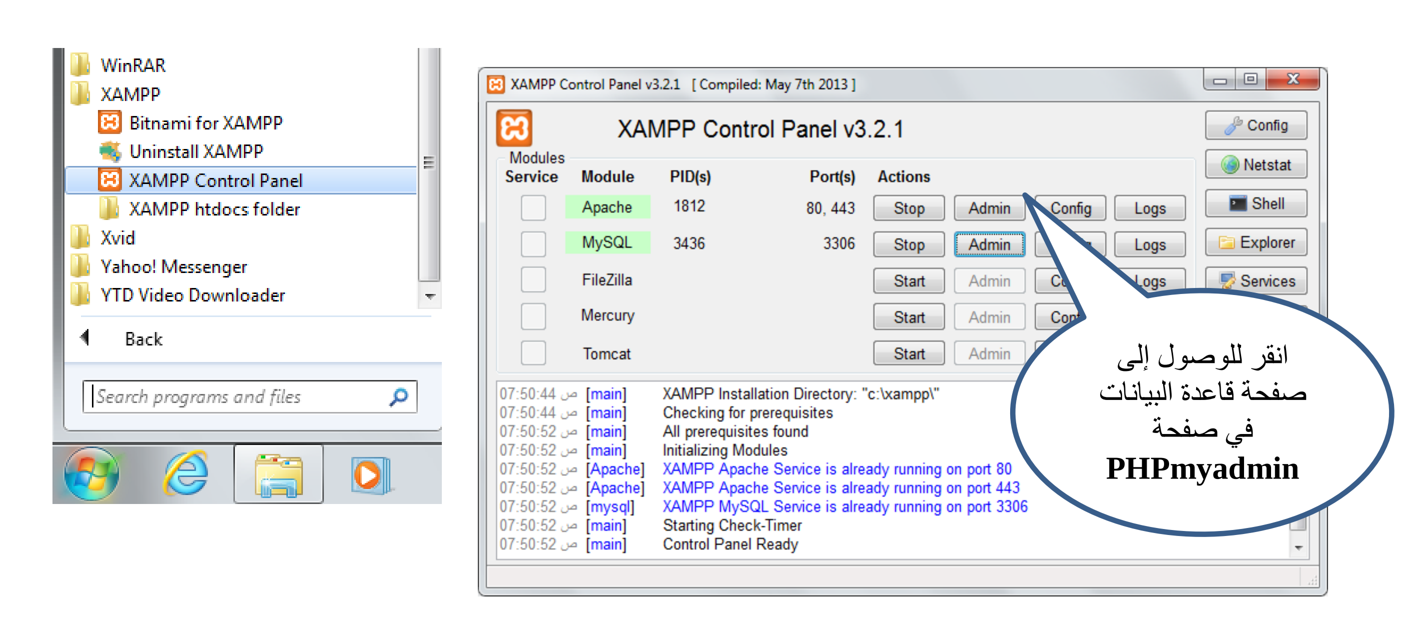
Table: Users

#	Name	Type	Length values	Index
1	Id	Integer	11	Primary
2	UseName	VarChar	100	
3	PassWord	VarChar	100	

شكل (٥٥) تصميم جدول Users

إنشاء جدول **users** (كما تعلمت انشاء الجداول بالوحدة الثانية) بقاعدة البيانات **dbdictionary** كما يلى:

تشغيل **XAMP control panel** من قائمة **start** كما بالشكل:



شكل (٥٦) تشغيل XAMP control panel

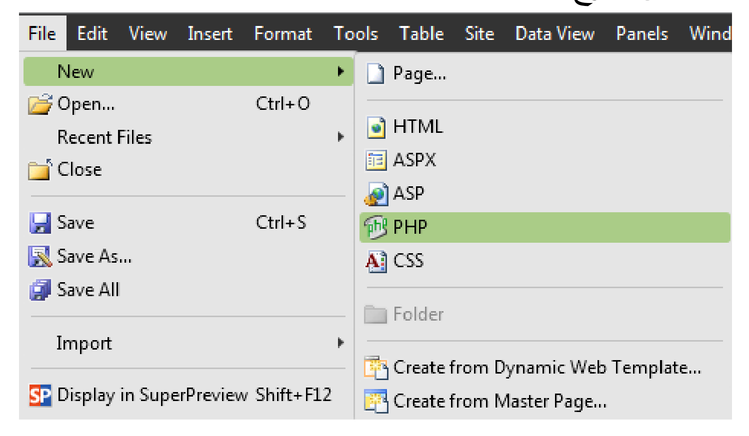
يبدو الجدول كما بالشكل بعد انشاؤه:



شكل (٥٧) جدول Users في قاعدة البيانات

ثانيًا: تصميم صفحة تسجيل مستخدم جديد

- افتح موقع **Dictionary** للتعديل باستخدام برنامج **Expression Web**.
- أنشئ صفحة جديدة من النوع **PHP** كما بالشكل:



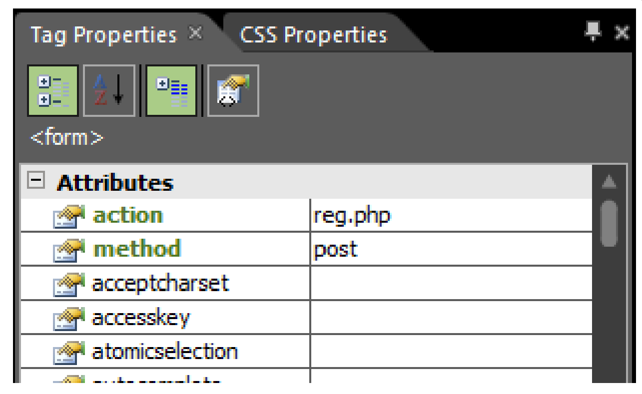
شكل (٥٨) إنشاء صفحة PHP في تطبيق Expression Web

- ادرج **Form** وضع عليه الكائنات objects كما بالشكل:

تسجيل مستخدم جديد
form
اسم المستخدم
كلمة المرور
تأكيد كلمة المرور
تسجيل

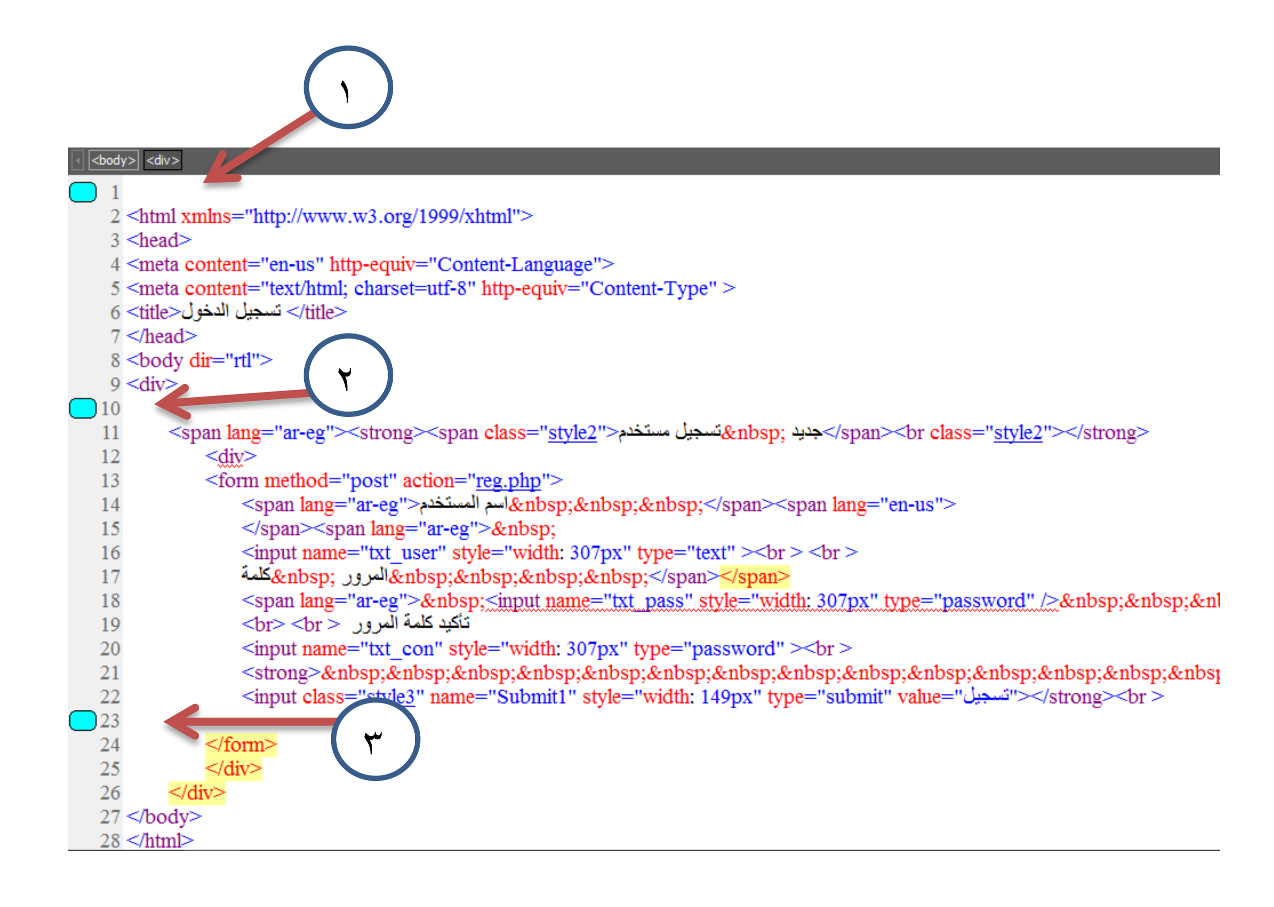
شكل (٥٩) إدراج نموذج يحتوى بعض عناصر التحكم في تطبيق Expression Web

في نافذة خصائص النموذج **Form** تأكد من تحديد القيمة **Post** للخاصية **Method** كما بالشكل:



شكل (٦٠) ضبط خصائص النموذج Form

- استعرض كود HTML يبدو كما بالشكل:

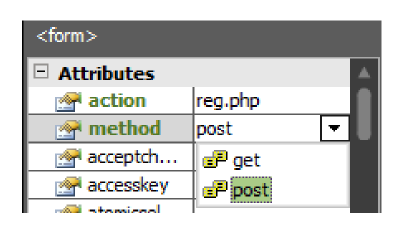


```
<body> <div>
1
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta content="en-us" http-equiv="Content-Language">
5 <meta content="text/html; charset=utf-8" http-equiv="Content-Type" >
6 <title>تسجيل الدخول </title>
7 </head>
8 <body dir="rtl">
9 <div>
10
11     <span lang="ar-eg"><strong><span class="style2">تسجيل مستخدم  جديد</span><br class="style2"></strong>
12         <div>
13         <form method="post" action="reg.php">
14             <span lang="ar-eg">اسم المستخدم   </span><span lang="en-us">
15             </span><span lang="ar-eg"> 
16             <input name="txt_user" style="width: 307px" type="text" ><br > <br >
17             كلمة  المرور    </span></span>
18             <span lang="ar-eg"> <input name="txt_pass" style="width: 307px" type="password" />  &nl
19             <br> <br > تأكيد كلمة المرور
20             <input name="txt_con" style="width: 307px" type="password" ><br >
21             <strong>              &nbs
22             <input class="style3" name="Submit1" style="width: 149px" type="submit" value="تسجيل"></strong><br >
23
24         </form>
25         </div>
26     </div>
27 </body>
28 </html>
```


شكل (٦١) كود HTML الخاص بإنشاء صفحة تسجيل مستخدم جديد

لاحظ أنه تم كتابة كود HTML بمجرد إنشاء النموذج من خلال نمط التصميم حيث تشير الأرقام إلى أماكن كتابة **PHP**.

الفرق بين Post و Get عند إنشاء النموذج Form:



في نافذة خصائص النموذج **Form** تم من تحديد القيمة **Post** للخاصية **Method** وكان يمكن تحديد القيمة **Get**

شكل (٦٢) ضبط الخاصية Method للنموذج Form

وإذا نظرت إلى كود **html** ستجد بداية **Tag** الخاص بالنموذج **Form** كالآتي:

```
<form method="post" action="reg.php">
```

يجب معرفة كيف يعمل النموذج form الذى قمت بادراجه:

النموذج Form يستخدم لتمرير أو إرسال كافة بيانات النموذج الموجودة في عناصر التحكم من مستعرض الويب إلى الخادم **Web Server** ويوجد طريقتين لإرسال بيانات النموذج وهما:

١- **<form method="GET">**

٢- **<form method="POST">**

وعند النقر على زر **Submit** يقوم بروتوكول **HTTP** بطلب **Request** لإرسال بيانات النموذج إلى الخادم **Web Server** ويعتمد الطلب **request** على الطريقة المتبعة إن كانت **Get** أم **Post**، وفي حالة القيمة **POST** فإن البيانات التي يتم إرسالها تكن مؤمنة وسرية.

بينما تستخدم **GET** في بعض الحالات منها:

- إذا كان حجم البيانات صغير.
- إذا كانت البيانات غير سرية لأنها تظهر في شريط عنوان صفحة مستعرض الإنترنت.

يمكنك عزيزى الطالب اكتشاف الفرق بين GET و POST عند انشاء نموذج بيانات، وتحديد ايهما أفضل لضمان أمان الموقع ، وذلك من خلال تنفيذك لنشاط ٢ ، بكتاب الأنشطة والتدريبات ص

- اضف كود **PHP** مكان الأرقام المشار اليها في نمط كود **HTML** كما يلي:

الرقم	كود PHP
١	`<?php` `session_start();` `?>`
٢	`<?php` `include("header.php");` `?>`

```
<?php
include("connection.php");

if(isset($_POST['Submit1']))
    {
    $txt_user=$_POST['txt_user'];
    $txt_pass=$_POST['txt_pass'];
    $txt_con=$_POST['txt_con'];

        mysql_query("SET NAMES 'utf8'");
        $query=mysql_query("insert into users values('','$txt_user','$txt_pass')");

    }
?>
```

٣

شكل (٦٣) كود PHP الخاص بإنشاء صفحة تسجيل مستخدم جديد

شرح الكود السابق:

كود PHP رقم ١

الإعلان عن استخدام **Session** داخل كود الصفحة.

```
<?php
session_start();
?>
```

مفهوم Session: تعني "جلسة" وهو طريقة لتخزين بيانات في متغير داخل ذاكرة الخادم Server لمستخدم واحد فقط ويكون هذا المتغير متاحًا للاستخدام خلال كافة صفحات الموقع، وذلك طيلة الوقت الذي يظل فيه المستخدم يتجول في صفحات الموقع.

فمثلًا: عند تصميم صفحة دخول مستخدم يتم التحقق من كون المستخدم قد قام بتسجيل دخول من عدمه حتى يتم استخدام متغيرات الجلسة **Session variables** خلال صفحات الموقع، وذلك من خلال إضافة الكود التالي في بداية أي صفحة بالموقع نريد استخدام متغيرات الجلسة بها.

```
<?php
session_start();
?>
```

شكل (٦٤) كود PHP الخاص بدء الـ Session

كود PHP رقم ٢

تضمين صفحة "**Header.php**" كما سبق شرحه.

```
<?php
include("header.php");
?>
```

كود PHP رقم ٣

- **تضمين صفحة الاتصال بقاعدة البيانات كما سبق شرحه.**

```
include("connection.php");
```

• استخدام المتغير $_POST للحصول على قيم عناصر التحكم في النموذج عند النقر على زر "Submit1".

تستخدم جملة IF للتحقق من الضغط على زر Submit1 حيث يتم تخصيص محتوى عناصر التحكم على النموذج للمتغيرات.

```
if(isset($_POST['Submit1']))
```

```
$txt_user=$_POST['txt_user'];
$txt_pass=$_POST['txt_pass'];
$txt_con=$_POST['txt_con'];
```

• استخدام دالة mysql_query والتي تساعد في حل مشكلة إظهار البيانات على صفحة المستعرض باللغة العربية بشكل صحيح وعدم ظهورها علامات استفهام هكذا ؟؟؟؟؟؟؟.

```
mysql_query("SET NAMES 'utf8'");
```

• من خلال الدالة mysql_qurey يتم إدراج قيم المستخدم من اسم وكلمة مرور في المتغيرين '$txt_user','$txt_pass'، وتخصص القيمة true للمتغير $query في حالة نجاح عملية الإدراج في جدول قاعدة البيانات.

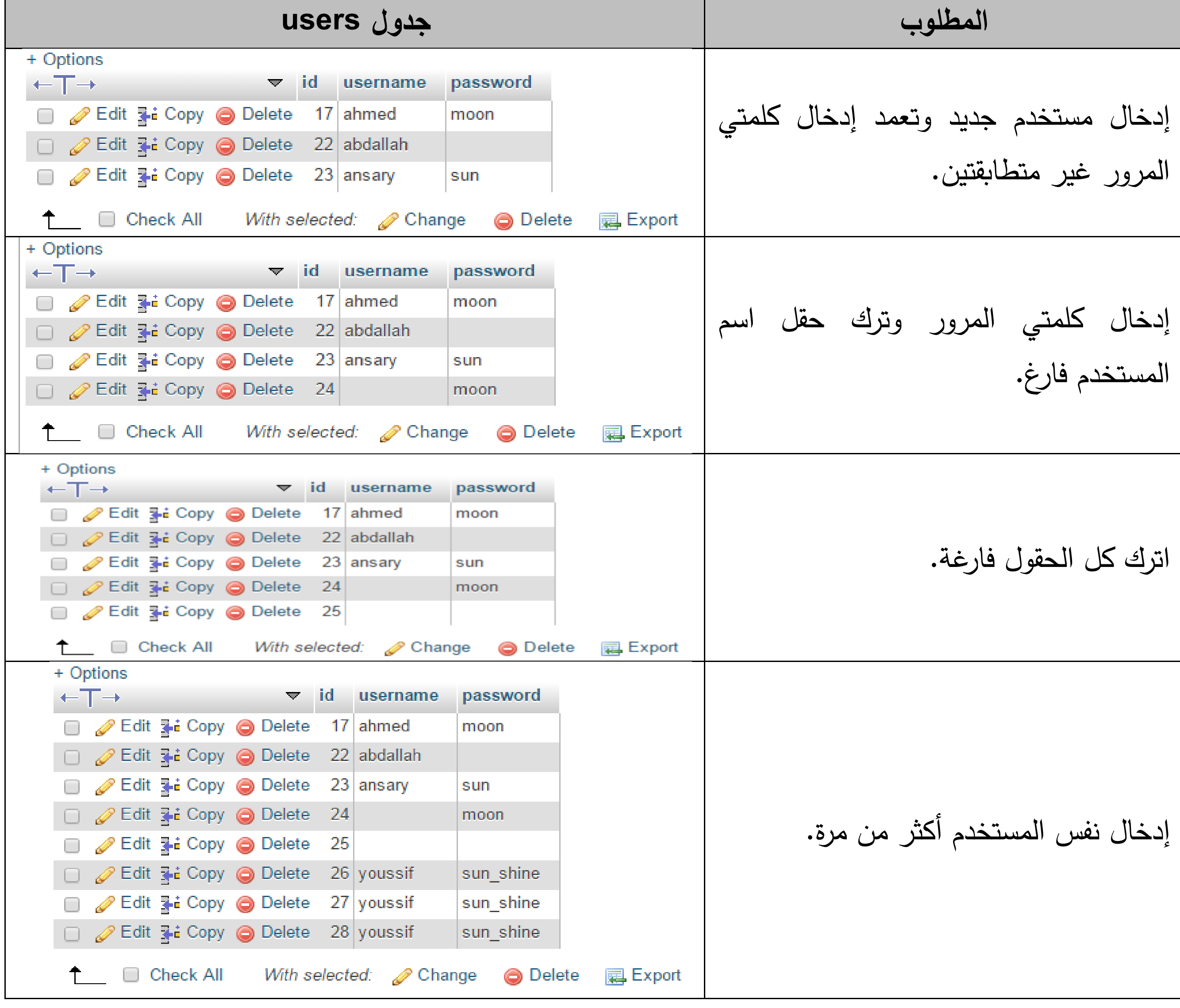


ملاحظة: جملة SQL "Insert Into" تستخدم لإضافة بيانات سجل جديد إلى جدول قاعدة البيانات Terms.

• افتح الصفحة في المستعرض ثم جرب الآتي مع فتح جدول Users في كل مرة:

http://localhost/dictionary/reg.php

المطلوب	جدول users
إدخال مستخدم جديد وإدخال كلمتي المرور متطابقتين ثم النقر على زر تسجيل، ثم افتح جدول Users وانظر هل تم إضافة البيانات أم لا؟	+ Options id | username | password Edit Copy Delete | 17 | ahmed | moon Check All With selected: Change Delete Export
إدخال مستخدم جديد وترك حقول كلمة المرور فارغة.	+ Options id | username | password Edit Copy Delete | 17 | ahmed | moon Edit Copy Delete | 22 | abdallah | Check All With selected: Change Delete Export



المطلوب	جدول users
إدخال مستخدم جديد وتعمد إدخال كلمتي المرور غير متطابقتين.	id / username / password 17 ahmed moon 22 abdallah 23 ansary sun
إدخال كلمتي المرور وترك حقل اسم المستخدم فارغ.	id / username / password 17 ahmed moon 22 abdallah 23 ansary sun 24 moon
اترك كل الحقول فارغة.	id / username / password 17 ahmed moon 22 abdallah 23 ansary sun 24 moon 25
إدخال نفس المستخدم أكثر من مرة.	id / username / password 17 ahmed moon 22 abdallah 23 ansary sun 24 moon 25 26 youssif sun_shine 27 youssif sun_shine 28 youssif sun_shine

شكل (٦٥) احتمالات إدخال اسم المستخدم وكلمة المرور

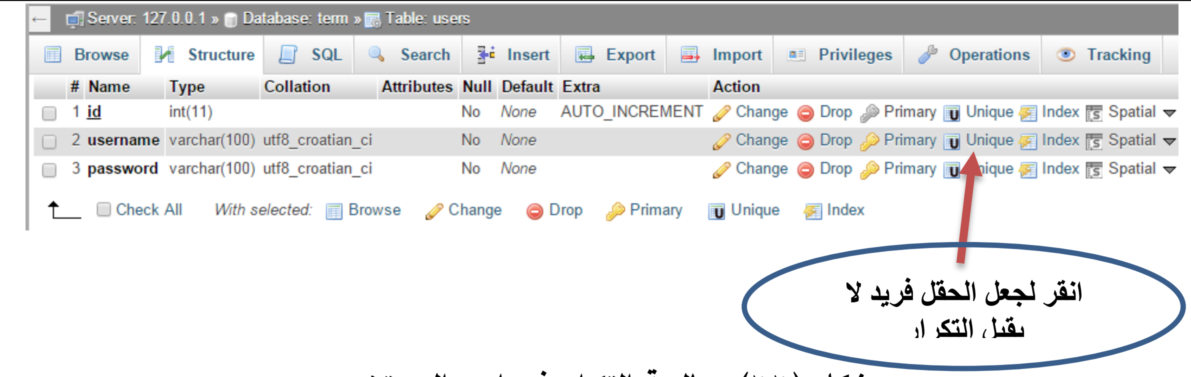
من البيانات بجدول **users يتضح أن النموذج form بهذا الشكل لا يوجد به أي مراعاة لقواعد التأمين وذلك للآسباب التالية:**

- لا يوجد تحقق من صحة البيانات المدخلة (قبول حقول فارغة – عدم اختبار تطابق كلمتي المرور).
- ما يتم تخزينه من كلمات المرور هو كلمة المرور الأولى بدون عمل مقارنة تطابق.
- حقل **username** بجدول **users** يقبل التكرار وهذا غير منطقي فلا يجب وجود أكثر من مستخدم بنفس الاسم.
- كلمة المرور ظاهرة بالجدول ويجب تشفيرها.

لمعالجة تلك المشاكل يتم اجراء الآتي:

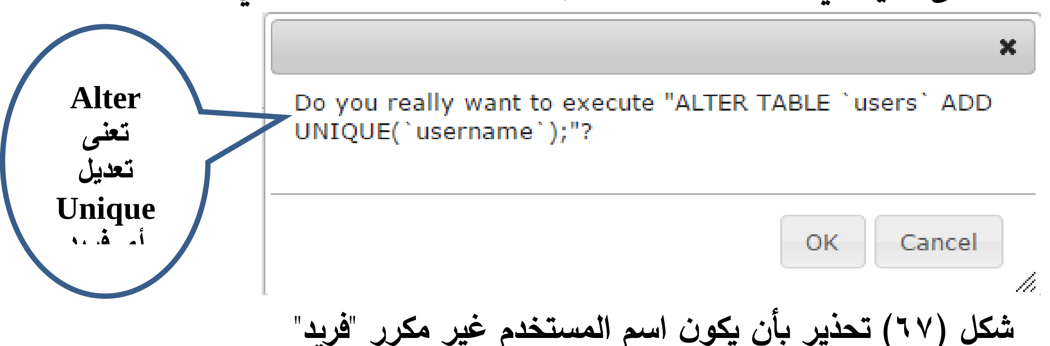
١– معالجة التكرار في حقل username بجدول users

انقر على **Structure** بصفحة **MYSQL** لعرض مواصفات الحقول للجدول **Users**



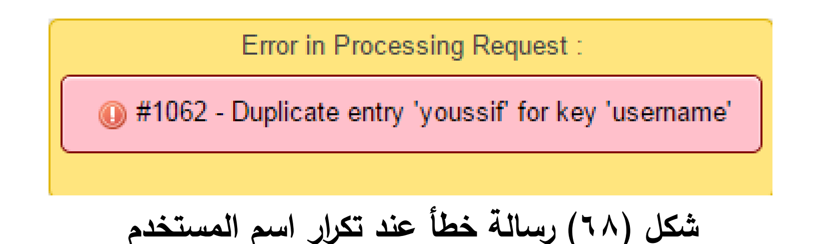
شكل (٦٦) معالجة التكرار في اسم المستخدم

بعد النقر على **Unique** وتعنى "فريد" أي لا يقبل التكرار، فتظهر النافذة كما بالشكل التالي:



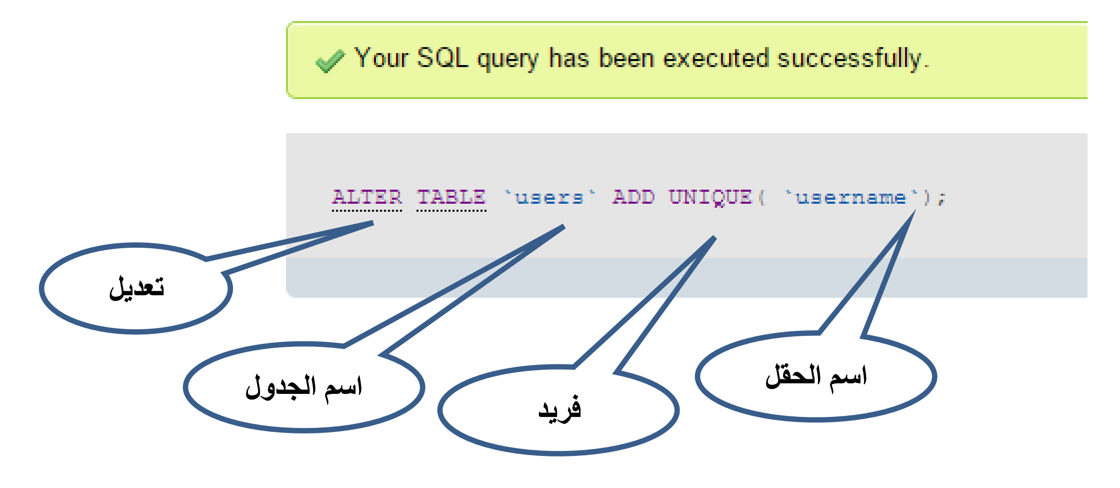
شكل (٦٧) تحذير بأن يكون اسم المستخدم غير مكرر "فريد"

انقر **OK**، وقد تظهر لك نافذة توضح عدم إمكانية جعل الحقل **Unique** لوجود سجل أو أكثر متكرر بالفعل، لذا يفضل ضبط خاصية **Unique** قبل إدخال البيانات.



شكل (٦٨) رسالة خطأ عند تكرار اسم المستخدم

عند نجاح هذه العملية يظهر لك الرسالة التالية ولا تظهر رسالة الخطأ السابقة:



شكل (٦٩) رسالة تفيد تشغيل جملة SQL بنجاح

٢- التحقق من عدم ترك أي حقل فارغ وأن كلمتي المرور متماثلتين بكود php والذي ينفذ على جهاز Server:

في هذه الحالة يتم كتابة جملة شرطية بعد تخصيص المتغيرات للتأكد من أن اسم المستخدم وكلمة المرور غير فارغين وكذلك كلمتي المرور متماثلتين.

```
<?php
    include("connection.php");
    if(isset($_POST['Submit1']))
        {
        $txt_user=$_POST['txt_user'];
        $txt_pass=$_POST['txt_pass'];
        $txt_con=$_POST['txt_con'];

        if($txt_user !== "" && $txt_pass !== "" && $txt_pass == $txt_con)
            {
            mysql_query("SET NAMES 'utf8'");
            $query=mysql_query("insert into users values('','$txt_user','$txt_pass')");
            }
        else
            {
            echo "<label id='Label1' style='color: #FF0000;font-size: large;'>يجب ادخال اسم المستخدم وكلمة المرور وأن تكون كلمتى المرور متكافئتين</label>";
            }
        }
?>
```

الجملة الشرطية المطلوبة

شكل (٧٠) كود PHP الخاص بالتأكد من عدم ترك أي حقل فارغ وتطابق كلمة المرور وتأكيدها

- التعبير الشرطي مع جملة **IF**

```
if($txt_user !== "" && $txt_pass !== "" && $txt_pass == $txt_con)
```

أى إذا كان المتغير **$txt_user** لا يساوي سلسلة حرفية فارغة و إذا كان المتغير **$txt_pass** لا يساوى سلسلة حرفية فارغة و إذا كان المتغير **$txt_pass** يساوي المتغير **$txt_con**، يصبح الناتج النهائي true حيث يجب التحقق من الثلاثة شروط المذكورة لكي يكون الناتج النهائي **True** وفي حالة عدم تحقق شرط أو أكثر منهم يصبح الناتج النهائي **False**.

- احفظ الصفحة.
- حمل الصفحة في مستعرض الإنترنت، ادخل بيانات مستخدم جديد مع ترك أحد الحقول فارغة ... سجل ملاحظتك.فى كتاب الانشطة والتدريبات صفحة
- سجل دخول مرة أخرى مع تعمد عدم مطابقة كلمتي المرور ...

٣- التحقق من أنه تم تسجيل البيانات في الجدول بنجاح

في هذه الحالة يتم اختبار المتغير **$query** والذي يكون ناتجه **True** عند تنفيذ جملة **SQL** بنجاح.

```
<?php
    include("connection.php");
    if(isset($_POST['Submit1']))
        {
        $txt_user=$_POST['txt_user'];
        $txt_pass=$_POST['txt_pass'];
        $txt_con=$_POST['txt_con'];
        if($txt_user !== "" && $txt_pass !== "" && $txt_pass == $txt_con)
            {
            mysql_query("SET NAMES 'utf8'");
            $query=mysql_query("insert into users values('','$txt_user','$txt_pass')");
            if($query)
                {
                echo("<label id='Label1' style='color: #FF0000;font-size: large;'>تم تسجيل البيانات بنجاح</label>");
                }
            else
                {
                echo("<label id='Label1' style='color: #FF0000;font-size: large;'>لم يتم اضافة البيانات</label>");
                }
            }
        else
            {
            echo "<label id='Label1' style='color: #FF0000;font-size: large;'>
            يجب ادخال اسم المستخدم وكلمة المرور وأن تكون كلمتى المرور متكافئتين</label>";
            }
        }
?>
```

الجملة الشرطية المطلوبة

شكل (٧١) كود PHP الخاص بالتأكد من تسجيل في جدول قاعدة البيانات بنجاح

٤- تشفير كلمة المرور

يوجد العديد من الأساليب لتشفير كلمة المرور ومثال على ذلك يمكن استخدام الدالة **MD5** وهي أحدى الدوال التي تستخدم في التشفير.

<u>الدالة MD5(Message-Digest Algorithm)</u>

تقوم هذه الدالة بإرجاع ناتج تشفير سلسة حرفية بأسلوب ١٢٨ بت.

مثال

كود PHP	الناتج في صفحة المستعرض
`<?php` `$str = "moon";` `echo md5($str);` `?>`	6d4db5ff0c117864a02827bad3c361b9

افتح كود صفحة **reg.php** واكتب ما هو مشار إليه على الشكل التالي في الأماكن الموضحة بدائرة على الكود التالي:



وبالتالي يصبح محتوى المتغير **$txt_pass** والمتغير **$txt_con** كلمة المرور المشفرة

احفظ الصفحة ثم حمل الصفحة في المستعرض.

– سجل أكثر من مستخدم جديد.

– افتح جدول **users.**



	id	username	password
Edit Copy Delete	30	Sama_Elansary	0...0e267a57e4dce30c85d5b09e751aa
Edit Copy Delete	31	Fatma_Abdallah	19f79239c0c369c73df66e3dd20dcf4f
Edit Copy Delete	32	Heba_Youssif	edda2fc163233f03ffba47c670d4117c
Edit Copy Delete	33	Youssif_Abdallah	7f9d3e4a2e6ff5e78ccaf346214ad919

شكل (٧٢) عرض كلمة المرور بعد التشفير في جدول قاعدة البيانات

وهناك طرق أخرى لتامين كلمة المرور أكثر من ذلك، ويمكنك عزيزي الطالب البحث عنها عبر الإنترنت.

كراسة الأنشطة: نشاط "التعلم الذاتي"

كراسة الأنشطة: نشاط "ابدأ من حيث انتهى الأخرون"

الموضوع الثالث

تصميم صفحة تسجيل الدخول login والخروج signout.php

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يصمم صفحة الدخول Login باستخدام Expression Web.

٢- يتعاون مع زملاءه فى كتابة كود HTML&PHP لإنشاء صفحة تسجيل الدخول.

٣- يصمم صفحة تسجيل الخروج Signout باستخدام Expression Web.

ثالثاً: تصميم صفحة الدخول login باستخدام Expression Web

فكرة هذه الصفحة هي:

١- إدخال اسم المستخدم وكلمة المرور والنقر على زر دخول.

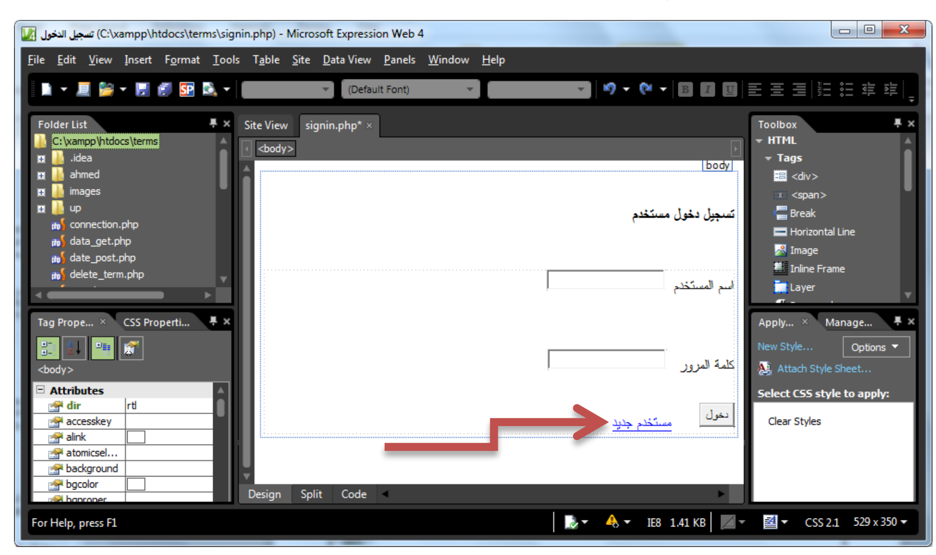
٢- البحث في جدول **users** عن اسم المستخدم وفي حالة وجوده يتم مقارنة كلمة المرور التي أدخلها المستخدم بكلمة المرور المخزنة في جدول قاعدة البيانات.

٣- في حالة التأكد من اسم المستخدم وكلمة المرور، يتم تخزين اسم المستخدم في المتغير الخاص بالجلسة **Session**، وعرض رسالة ترحيب باسم المستخدم.

٤- تأثر صفحات أخرى بالدخول مثل صفحة إضافة مصطلح وصفحة التعديل وصفحة الحذف حيث يجب تعديل الكود بهم بحيث يكون الاضافة والتعديل والحذف بعد قيام المستخدم بتسجيل دخول.

خطوات العمل

- افتح برنامج **Expression Web**، وانشئ صفحة باسم **Signin.php**.
- ادرج نموذج **Form** وضع عليه عناصر التحكم **Controls** كما بالشاشة التالية:



شكل (٧٣) تصميم واجهة صفحة تسجيل دخول مستخدم

لاحظ الارتباط التشعبي "مستخدم جديد" والمرتبط بصفحة مستخدم جديد **reg** السابق إنشاؤها حيث تبدو نافذة الكود كما بالشكل التالي:

١

```
Site View   signin.php ×
1
2 <html xmlns="http://www.w3.org/1999/xhtml">
3     <head>
4     <meta content="en-us" http-equiv="Content-Language">
5     <meta content="text/html; charset=utf-8" http-equiv="Content-Type" >
6         <title>تسجيل الدخول</title>
7     </head>
8
9 <body dir="rtl">
10 <?php
11 include("header.php");
12  ?>
13  <span lang="ar-eg"><strong><span class="style2">تسجيل دخول مستخدم </span>
14      <br class="style2">
15      </strong>
16 <div>
17    <form name="admin" action="signin.php" method="post"
18    enctype="multipart/form-data">
19    اسم المستخدم  <input name="user" type="text"><br><br><br >
20   كلمة المرور   <input name="pass" type="password" ><br>
21    <br >
22   <input name="submit1" type="submit" value="دخول">
23         <a href="reg.php"> مستخدم جديد  </a>
24    </form>
25
26    </div>
27 </body>
28 </html>
Design   Split   Code
```

كود PHP الخاص
بصفحة header

٢

شكل (٧٤) كود PHP , HTML الخاص بإنشاء صفحة تسجيل الدخول

اضف كود **php** التالي مكان الأرقام المشار إليها في شاشة الكود السابقة:

الرقم	الكود
١	`<?php session_start(); ?>`
٢	(الكود أدناه)

```
<?php
  include("connection.php");
  mysql_query("SET NAMES 'utf8'");

  if(isset($_POST['submit1']))          // اختبار النقر على زر Submit
   {
   $usern=$_POST['user'];
   $passw=$_POST['pass'];
   $passw=md5( $passw);

   if($usern !== "" &&   $passw !== "" )      // التأكد من إدخال الاسم وكلمة المرور
     {
     $sql="select * from users where username='$usern'  &&  password= '$passw' ";
     $query=mysql_query($sql);
     $num=mysql_num_rows($query);

     if ($num == 1)      // التأكد من وجود اسم المستخدم بالجدول
        {
        $row=mysql_fetch_array($query);
        $_SESSION['username']=$row['username'];
        header("Location: index.php");
        exit;
        }

     }

   }
?>
```

شكل (٧٥) كود PHP الخاص بإنشاء صفحة تسجيل الدخول

شرح الكود السابق

كود PHP رقم ١

في بداية صفحة تسجيل الدخول نجد الكود الذي يُعلن عن بدء جلسة **Session** للمستخدم، وذلك لأن أي مستخدم جديد للموقع ينبغي أن يدخل باسم مستخدم **Username** وكلمة مرور **Password**.

كود PHP رقم ٢

- أول سطرين لتضمين صفحة الاتصال بقاعدة البيانات والتي سبق شرحها، وأيضًا استخدام دالة **mysql_query** والتي تساعد في حل مشكلة إظهار البيانات على صفحة المستعرض باللغة العربية بشكل صحيح وعدم ظهورها علامات استفهام هكذا ؟؟؟؟؟؟، وفي باقي الكود:
- تستخدم جملة **IF** في اختبار الضغط على زر **Submit** بالدالة **Isset** فإذا وعند تحقق الشرط في جملة **IF:** ينفذ سطري الكود التاليين لجملة **IF**، حيث يتم إنشاء المتغير **$Usern** ويخصص له اسم المستخدم الذي تم إدخاله في عنصر التحكم **User** على الـ **Form**، وأيضًا يتم إنشاء المتغير **$Passw** ويخصص له كلمة المرور التي تم إدخالها في عنصر التحكم **Pass** على الـ **Form**.

- لاحظ إضافة السطر `$passw=md5( $passw);` لتشفير كلمة المرور المدخلة بواسطة المستخدم حتى يتم مقارنتها بما هو مخزن بجدول users المشفرة أيضًا.
- وفي جملة **IF** الثانية يتم التحقق من أن عنصري التحكم اسم المستخدم وكلمة المرور غير خاليين، حيث نسأل في جملة الشرط **IF** من خلال شرط مركب إن كان **$usern** لا يساوي "= =!" قيمة فارغة أي **Null** وكذلك أن متغير كلمة المرور ليس خاليًا، مع ملاحظة أنه تم استخدام المعامل "**&&**" والذي يعني **AND** للربط بين شرطين، فإذا تحقق الشرطين معًا وكانت قيمة كلٍ من اسم المستخدم وكلمة المرور غير خاليين **Null** أي فيهما قيمة، فيتم تنفيذ الكود التالي:
- من خلال جملة **Select** يتم البحث في جدول User في قاعدة البيانات عن اسم المستخدم الذي تم إدخاله في عنصر التحكم على **Form** ووضع في المتغير **$Usern** في حقل **Username** في قاعدة البيانات، وأيضًا يتم البحث عن كلمة المرور التي تم إدخالها في عنصر التحكم على **Form** ووضعت في المتغير **$passw** في حقل **Password** في قاعدة البيانات، ويتم وضع نتيجة البحث في المتغير **$sql** شريطة أن يكون الاسم وكلمة المرور معًا.
- يتم تخصيص ناتج تنفيذ الدالة **Mysql_query** على المتغير **$sql**، للمتغير **$query**.

الدالة **Mysql_num_rows** تستخدم في تحديد عدد السجلات التي تم الحصول عليها في الدالة ووضعها في المتغير **$sql**، ويتم تخزين نتيجة الدالة في المتغير **$sum**.

- في الجزء الأخير من صفحة تسجيل دخول المستخدم، يتم اختبار المتغير **$num** إذا كان قيمته تساوي ١ أي أن هناك مستخدم واحد فقط يوجد اسمه وكلمة المرور في جدول **Users**، فإذا تحقق الشرط ووجد سجل واحد فقط، يتم تنفيذ الدالة **mysql_fetch_array** وتحول نتيجة جملة **sql** إلى مصفوفة يكون نتيجتها قيم سجل تُخزن في المتغير **$row** وتنقل المؤشر على السجل التالي في جدول **Users** في قاعدة البيانات.

ثم يتم تخصيص اسم المستخدم من حقل **Username** في قاعدة البيانات، للمتغير **username** في الجلسة **Session**، وبالتالي يمكن مقارنتها والتحقق منها في أي صفحة من صفحات الموقع، ويطبعه على مستعرض الإنترنت، ثم يتم الانتقال إلى الصفحة الرئيسة "**Index.php**"، ويتم الخروج من جملة **IF** بالكود **Exit**، ثم يتم إغلاق جمل **IF** الثلاثة بالأقواس الثلاثة **{**.

رابعًا: تصميم صفحة تسجيل الخروج signout.php باستخدام Expression Web

تعتمد فكرة صفحة تسجيل خروج على تخصيص القيمة null إلى المتغير $_SESSION["]USERNAME واعادة التوجيه إلى الصفحة الرئيسة INDEX.PHP وهو ما يظهر في الكود:

```
<?php
session_start();
 $_SESSION['username'] = "";
 header("Location: index.php");
 exit;
?>
```

الموضوع الرابع

إجراءات تأمين الموقع

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يجري عدة تعديلات فى الموقع

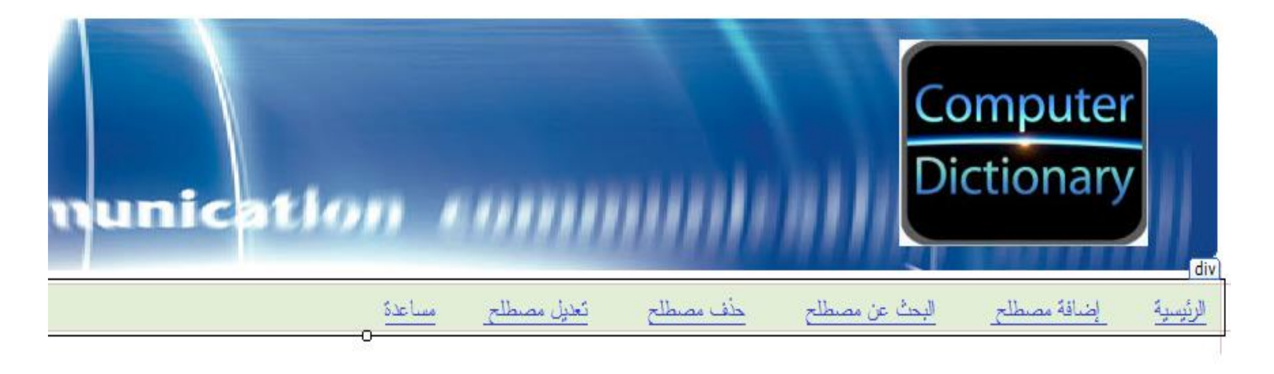
٢- يمارس عمليات الربط التشعبي

٣- يوظف جملة الشرط IF فى التحقق من تسجيل الخروج

١- اضافة كود بدء الجلسة في بداية كل صفحة يستخدم بها الجلسة.

```
<?php
 session_start();
?>
```

٢- تعديل الكود في صفحة header



شكل (٧٦) صفحة Header المراد إضافتها إلى صفحة تسجيل دخول

المطلوب اختبار هل تم تسجيل الدخول أم لا

في حالة تسجيل الدخول يتم الآتي:

- إضافة عبارة مرحبًا بكم وبجوارها "اسم المستخدم".
- عرض إرتباط تشعبي باسم تسجيل الخروج.

في حالة عدم الدخول يضاف ارتباط تشعبي باسم تسجيل الدخول

```
Site View | header.php* | headerc.php

1  <html  dir="rtl">
2  <head>
3  <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
4  </head>
5  <body>
6  <div>
7  <img  height='161' src='images/banner.jpg' width='1267'>
8  </div>
9  <div  style="width: 1256px;text-align:right"    >
10   <a href="index.php">الرئيسية</a>
11       <a href="input term terms.php"> إضافة مصطلح
12 </a>         <a href="search text.php">البحث عن مصطلح</a>
13          <a href="delete term.php">حذف مصطلح</a>
14          <a href="update term.php">تعديل مصطلح
15 </a>       <a href="Help.php">مساعدة </a>
16 
17 </div>
18 </body>
19 
20 </html>

Design | Split | Code
```

اضف كود php هنا

شكل (٧٧) التعديل في صفحة Header

```
<?php
 if(@$_SESSION['username'] == "")
     {
     echo ("  <a   href='signin.php'><span lang='ar-eg'>تسجيل الدخول</span> </a>");
     echo ("        ");
     echo "مستخدم غير مسجل";
     $_SESSION['username'] = "";
     }
else
     {
     echo ("  <a   href='signout.php'><span lang='ar-eg'>تسجيل الخروج</span></a>");
     echo ("        ");
     echo("مرحبا بك". "   " . $_SESSION['username']);
     }
?>
```

شكل (٧٨) إضافة الارتباط التشعبي "تسجيل دخول" من خلال كود PHP في صفحة Header

الجملة الشرطية IF

if(@$_SESSION['username'] == "")

- تتعامل لغة **PHP** مع العلامة **@** على أنها متغير، لو كان بدون قيمه لا يظهر رسالة خطأ.
- تذكر من قواعد لغة **PHP** وضع علامة **$** قبل اسم المتغير.
- **$_Session** متغير في ذاكرة جهاز الخادم للتحقق من كون المستخدم سجل دخول أم لا.
- الشرط الخاص بجملة **IF** هو: **['username']= = ""** أي هل اسم المستخدم يساوي **"Null"** أي خالي لم يدخل فيه بيانات وهناك حالتين إما أن يتحقق الشرط أو لا يتحقق:
  - **فإذا تحقق الشرط "Yes"**: يجعل عبارة "تسجيل دخول" ارتباط تشعبي لصفحة التسجيل المسماة "**signin.php**" ويترك عدة مسافات ويطبع على صفحة المستعرض أن "المستخدم غير مسجل"، ويجعل متغير **Session** قيمة **Null** بالكود: **$_SESSION['username'] == "";**.

- وإذا لم يتحقق الشرط "No" أي أن **Session** يحتوي على قيمة أي اسم للمستخدم: يجعل عبارة "تسجيل الخروج" ارتباط تشعبي لصفحة الخروج والمسماة "**signout.php**" ويترك عدة مسافات ويطبع على صفحة المستعرض رسالة ترحيب، "مرحبًا بكم" ثم يترك عدة مسافات، ويكتب اسم المستخدم الذي قام بإدخاله في متغير **Session** وذلك بالكود:

("مرحبا بكم". " " . $_SESSION['username']);

لاحظ أنه يمكن استخدام كود **Echo** لطباعة أكثر من ثابت أو متغير على صفحة المستعرض وذلك بالفصل بينها بنقطة " . ".

وأخيرًا تم إغلاق أو وضع نهاية جملة **If** "**}**" وعلامة النهاية للجزء الأول **echo ("</div>");** ولاحظ إضافة **Echo** لأن كود **div** الخاص بلغة **HTML** داخل كود **PHP** وأخيرًا نهاية كود PHP "**?>**".

٣-تعديل الكود في صفحة update

اضف جملة شرطية في المكان المشار إليه لاختبار هل تم الدخول أم لا وفي حالة عدم تسجيل الدخول نعيد التوجيه إلى صفحة تفيد بعدم وجود صلاحية التعديل والخروج من الصفحة للتعديل.

```
<?php
 session_start();
?>
<html  dir="rtl">
<head>
<meta content="text/html; charset=utf-8" http-equiv="Content-Type" >
<title>تحديث مصطلح </title>
</head>

<body>

<?php

include("header.php");

?>

 <h1><p > تعديل مصطلح </p>    </h1>
<?php
 include("connection.php");
 mysql_query("SET NAMES 'utf8'");
$sql="select * from terms";
$query=mysql_query($sql);
$num=mysql_num_rows($query);
```


شكل (٧٩) إضافة كود PHP في صفحة التعديل يمسح للمستخدم الذي سجل دخول أن يقوم بالتعديل

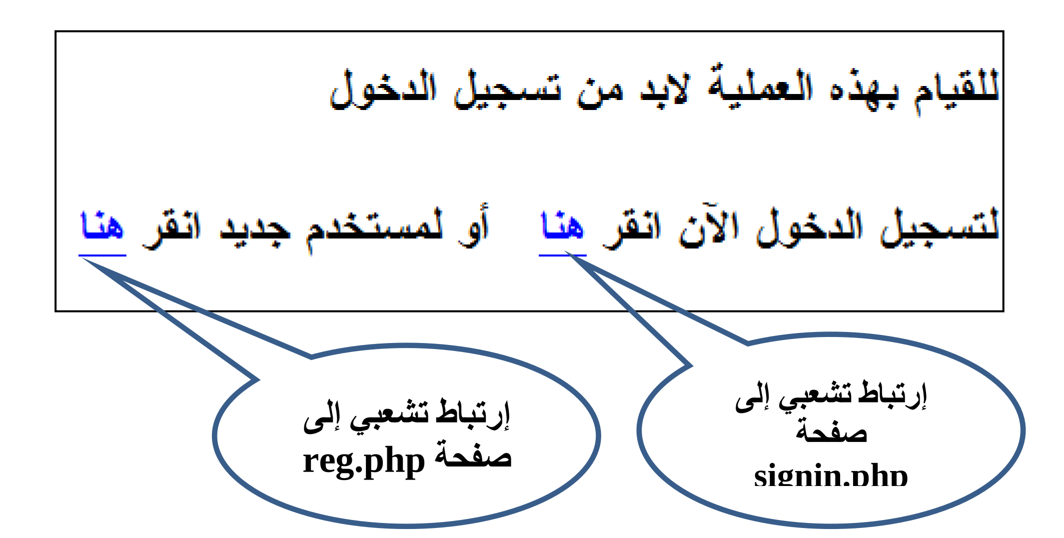
والجملة الشرطية كما يلي:

```
if($_SESSION['username'] == "")
        {
      header("Location: ask_to_sign_in.php");
         exit;
        }
```

شكل (٨٠) إضافة كود PHP في صفحة التعديل يعرض صفحة تطلب من المستخدم تسجيل دخول

تعني اذا لم يتم تسجيل الدخول يتم توجيه المستخدم إلى صفحة تسمى **ask_to_sign_in.php**، ثم الخروج من كود الصفحة.

صفحة **ask_to_sign_in.php** هي صفحة عادية تعرض رسالة توجه المستخدم لما يجب القيام به كما يلى:



٤- يتم إضافة نفس الجملة الشرطية السابقة في بداية كود PHP في صفحة إضافة مصطلح وأيضًا في صفحة حذف مصطلح.

وبذلك يسمح فقط بالإضافة والتعديل والحذف للمستخدم الذي قام بتسجل دخول.

الموضوع الخامس

تأمين قاعدة البيانات mysql

فى نهاية الموضوع يتوقع أن يكون الطالب قادرًا على أن:

١- يذكر طرق تأمين قاعدة البيانات

٢- يناقش اكبر عدد من الأفكار لتأمين قواعد البيانات.

٣- يعدل اسم المستخدم وكلمة المرور فى قاعدة البيانات.

وذلك بعمل التعديلات اللازمة في الموقع بما يتوافق مع أسلوب تسجيل دخول مستخدم كما يلي:

عند تثبيت install برنامج **XAMPP** يتم إنشاء مستخدم باسم **root** بدون كلمة مرور وهذا يعني السماح للجميع بالتعامل مع قاعدة البيانات ويجب عمل كلمة مرور لتأمين قاعدة البيانات.

تذكر الكود بصفحة connection

```
<?php
    $username="root";

    $password="";

    $database="term";

    $server="localhost";

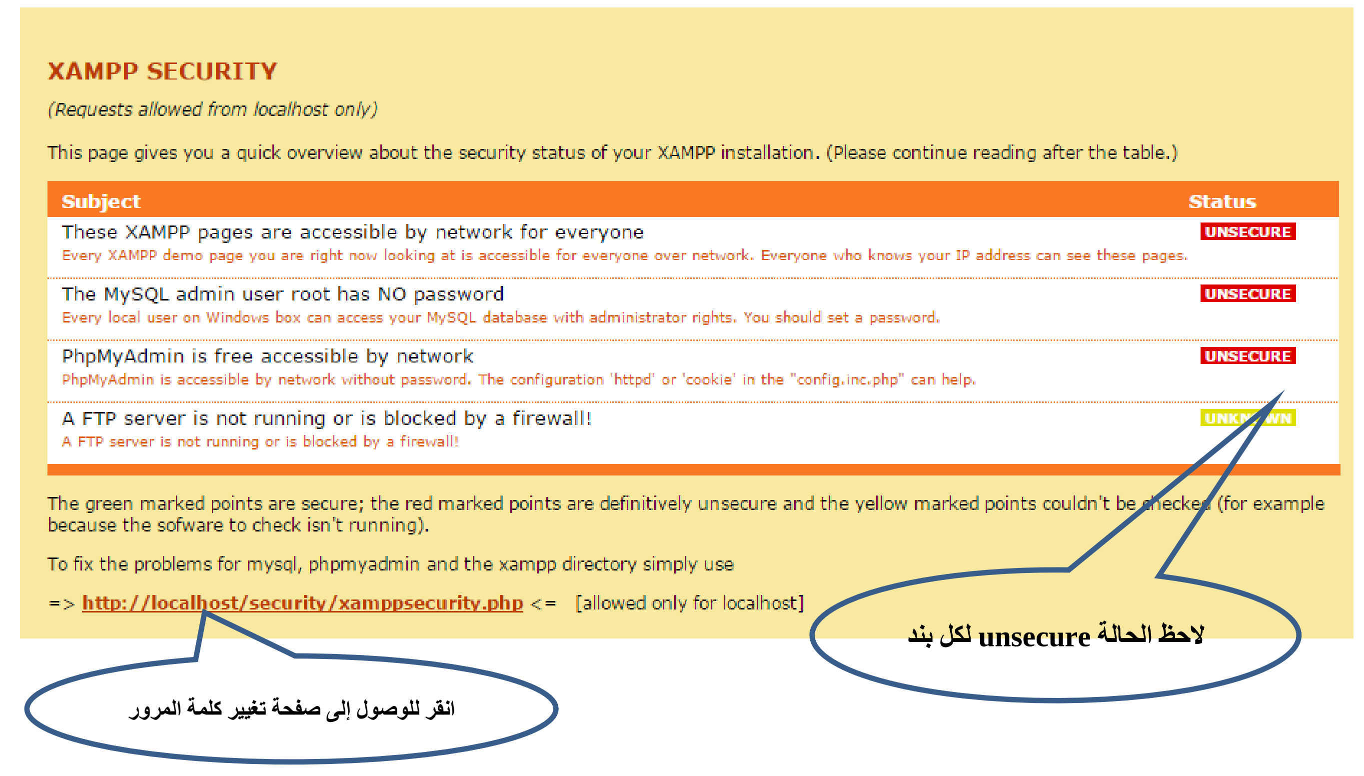
    $connect=mysql_connect("$server","$username","$password");
    if ($connect)
        {
        $select=mysql_select_db("$database") or die("هناك مشكلة فى قاعدة البيانات");
        }
        else
        {
        echo("لم يتم الاتصال بقاعدة البيانات");}
        }
?>
```

كلمة المرور فارغة

شكل (٨١) كود PHP في صفحة الاتصال بقاعدة البيانات بتأمين ضعيف

حمل صفحة التأمين بكتابة العنوان التالي في صفحة مستعرض الإنترنت:

http://localhost/security/



شكل (٨٢) تغيير اسم المستخدم كلمة المرور الخاصة بقاعدة البيانات

ومن خلال الصفحة بالشكل التالي، اكتب كلمة المرور الجديدة وتأكيدها، ثم انقر زر **Password Changing**

Security console MySQL & XAMPP directory protection

MYSQL SECTION: "ROOT" PASSWORD

MySQL SuperUser: **root**

New password:

Repeat the new password:

PhpMyAdmin authentification: *http* ○ *cookie* ◉

---- Security risk! ----

Safe plain password in text file? ☐

(File: C:\xampp\security\security\mysqlrootpasswd.txt)

[Password changing]

شكل (٨٣) إدخال كلمة مرور جديدة وتأكيدها لقاعدة البيانات أكثر أمانًا

بالطبع يجب تعديل كلمة المرور في صفحة **connection.php** كما يلي:

```
$password="HowTo_Design_php_web_site";
```

والتـــي يفضـــل أن تكـــون كلمـــة مـــرور صـــعبة ومعقـــدة حيـــث لـــن يـــتم اســـتخدامها إلا فـــي صـــفحة **connection.php** وبالتالي يكون من الصعب تخمين أو معرفة كلمة المرور على المخترق.